BIG-IP® Local Traffic Management
設定ガイド

バージョン 10.0.0

MAN-0292-00

製品バージョン

本マニュアルは、BIG-IP® Local Traffic Manager のバージョン 10.0.0 に適用されます。

発行日

本マニュアルは、2009 年 2 月 25 日に発行されました。

利用規約

著作権

Copyright 2009, F5 Networks, Inc. All rights reserved.

F5 Networks, Inc.（F5）では、本マニュアルに記載されている情報を正確かつ信頼性のあるものであると考えています。ただし、F5 は、この情報の使用、あるいはこの情報の使用から生じる第三者の特許または他の権利の侵害に関していかなる責任も負いません。該当するユーザライセンスにより明確に規定される場合を除き、F5 のすべての特許権、著作権、その他の知的財産権の下、明示的または暗黙的なライセンスを一切付与しません。F5 は、予告なしにいかなる時点でも仕様を変更する権利を保有します。

商標

F5、F5 Networks、F5 のロゴ、BIG-IP、3-DNS、Acopia、Acopia Networks、Application Accelerator、Ask F5、Application Security Manager、ASM、ARX、Data Guard、Enterprise Manager、EM、FirePass、FreedomFabric、Global Traffic Manager、GTM、iControl、Intelligent Browser Referencing、Internet Control Architecture、IP Application Switch、iRules、Link Controller、LC、Local Traffic Manager、LTM、Message Security Module、MSM、NetCelera、OneConnect、Packet Velocity、SSL Accelerator、SYN Check、Traffic Management Operating System、TMOS、TrafficShield、Transparent Data Reduction、uRoam、VIPRION、WANJet、WebAccelerator、および ZoneRunner は、米国および他の国における F5 Networks, Inc. の商標または登録商標で、F5 から明示的な書面による同意を受けていない使用は禁止されています。

特許権

本製品は米国特許番号 6,374,300; 6,473,802; 6,970,933; 7,051,126; 7,102,996; 7,146,354; 7,197,661; 7,206,282; 7,287,084 で保護されています。そのほかの特許は申請中です。

輸出規制に関する通知

本製品は、暗号化ソフトウェアを含む場合があります。輸出管理法（Export Administration Act）に基づき、本製品の米国外への輸出は、米国政府から違法行為とみなされる可能性があります。

無線周波数妨害に関する警告

本製品は Class A 製品です。米国内の環境では、本製品によって電波障害が発生する可能性があります。その場合は適切な対処策が必要です。

FCC への準拠

本製品は、FCC 規則の第 15 章に定められている Class A デジタルデバイスに関する規制要件に基づいて所定の試験を実施し、その適合性が認定されています。これらの制限は、本製品が商用環境で動作する際の有害な無線周波数妨害に対して適切な保護機能を提供することを目的としています。このユニットは、無線周波数エネルギーを発生および使用し、場合によっては放射する可能性があります。そのため、取扱説明書に従った設置や使用を行わない場合、無線通信に対して有害な妨害を生じることがあります。住宅地域で本製品を使用すると無線周波妨害が発生する可能性があります。その場合、ユーザは妨害状態を回避するために必要な手段を講じる必要があります。

製造元からの明示的な承認を受けずに本製品に変更を加えた場合、FCC 規則の第 15 章に基づいて本製品を操作するユーザの権限が無効になることがあります。

謝辞

本製品には、Bill Paul により開発されたソフトウェアが含まれています。

本製品には、Jonathan Stone により開発されたソフトウェアが含まれています。

本製品には、Manuel Bouyer により開発されたソフトウェアが含まれています。

本製品には、Paul Richards により開発されたソフトウェアが含まれています。

本製品には、NetBSD Foundation, Inc. およびその貢献者により開発されたソフトウェアが含まれています。

本製品には、Politecnico di Torino およびその貢献者により開発されたソフトウェアが含まれています。

本製品には、Swedish Institute of Computer Science およびその貢献者により開発されたソフトウェアが含まれています。

本製品には、University of California, Berkey およびその貢献者により開発されたソフトウェアが含まれています。

本製品には、Lawrence Berkeley Laboratory の Computer System Engineering Group により開発されたソフトウェアが含まれています。

本製品には、NetBSD プロジェクト用に Christopher G. Demetriou により開発されたソフトウェアが含まれています。

本製品には、Adam Glass により開発されたソフトウェアが含まれています。

本製品には、Christian E. Hopps により開発されたソフトウェアが含まれています。

本製品には、Dean Huxley により開発されたソフトウェアが含まれています。

本製品には、John Kohl により開発されたソフトウェアが含まれています。

本製品には、Paul Kranenburg により開発されたソフトウェアが含まれています。

本製品には、Terrence R. Lambert により開発されたソフトウェアが含まれています。

本製品には、Philip A. Nelson により開発されたソフトウェアが含まれています。

本製品には、Herb Peyerl により開発されたソフトウェアが含まれています。

本製品には、NetBSD プロジェクト用に Jochen Pohl により開発されたソフトウェアが含まれています。

本製品には、Chris Provenzano により開発されたソフトウェアが含まれています。

本製品には、Theo de Raadt により開発されたソフトウェアが含まれています。

本製品には、David Muir Sharnoff により開発されたソフトウェアが含まれています。

本製品には、SigmaSoft の Th. Lockert により開発されたソフトウェアが含まれています。

本製品には、NetBSD プロジェクト用に Jason R. Thorpe により開発されたソフトウェアが含まれています。

本製品には、And Communications（http://www.and.com）用に Jason R. Thorpe により開発されたソフトウェアが含まれています。

本製品には、NetBSD プロジェクト用に Frank Van der Linden により開発されたソフトウェアが含まれています。

本製品には、NetBSD プロジェクト用に John M. Vinopal により開発されたソフトウェアが含まれています。

本製品には、Christos Zoulas により開発されたソフトウェアが含まれています。

本製品には、University of Vermont および State Agricultural College と Garrett A. Wollman により開発されたソフトウェアが含まれています。

本製品には、Balazs Scheidler <bazsi@balabit.hu> により開発され、GNU Public License に基づいて保護されているソフトウェアが含まれています。

本製品には、Niels Mueller <nisse@lysator.liu.se> により開発され、GNU Public License に基づいて保護されているソフトウェアが含まれています。

次の文で、「このソフトウェア」とは Mitsumi CD-ROM ドライバを指しています。このソフトウェアは、Holger Veit および Brian Moore によって、「386BSD」および類似のオペレーティングシステム用に開発されました。「同様のオペレーティングシステム」には、「NetBSD」、「FreeBSD」、「Mach」（カーネギーメロン大学（CMU）開発）など、主として非営利の研究、教育用システムが含まれています。

本製品には、Apache Group によって Apache HTTP サーバプロジェクト（http://www.apache.org/）用に開発されたソフトウェアが含まれています。

本製品には、GNU Library General Public License（© 1998, Red Hat Software）（www.gnu.org/copyleft/lgpl.html）の下、Richard H. Porter から使用許諾されたソフトウェアが含まれています。

本製品には、Perl Artistic License（© 1997, 1998 Tom Christiansen and Nathan Torkington）の下で使用許諾された Perl ソフトウェアの標準版が含まれています。All rights reserved. 最新の Perl 標準版は、http://www.perl.com で入手できます。

本製品には、Jared Minch により開発されたソフトウェアが含まれています。

本製品には、OpenSSL Toolkit（http://www.openssl.org/）での使用を目的として、OpenSSL プロジェクトにより開発されたソフトウェアが含まれています。

本製品には、Eric Young（eay@cryptsoft.com）により作成された暗号化ソフトウェアが含まれています。

本製品には、GNU Public License で保護されている oprofile に基づくソフトウェアが含まれています。

本製品には、Tobi Oetiker 氏（http://www.rrdtool.com/index.html）が開発し、GNU General Public License に基づきライセンス付与されている RRDtool が含まれています。

本製品には、GNU General Public License（GPL）の許可を受けて、Dr. Brian Gladman から使用許諾されたソフトウェアが含まれています。

本製品には、Apache Software Foundation <http://www.apache.org/> により開発されたソフトウェアが含まれています。

本製品には、Hypersonic SQL が含まれています。

本製品には、Regents of the University of California、Sun Microsystems, Inc.、Scriptics Corporation、その他によって開発されたソフトウェアが含まれています。

本製品には、Internet Software Consortium により開発されたソフトウェアが含まれています。

本製品には、Nominum, Inc.（http://www.nominum.com）により開発されたソフトウェアが含まれています。

本製品には、Broadcom Corporation によって開発されたソフトウェアが含まれており、これは GNU Public License で保護されています。

本製品には、Lawrence Berkeley Laboratory の Computer System Engineering Group により開発されたソフトウェアが含まれています。Copyright © 1990-1994 Regents of the University of California.All rights reserved. 以下の条件を満たす場合、修正の有無を問わず、ソースおよびバイナリ形式での再配布および使用が許可されています

1. ソース・コードの再配布には上記の著作権表示、この条件リスト、および下記の免責条項を維持し明記する必要があります。

2. バイナリ形式での再配布には、配布と共に提供される文書および / または資料に上記の著作権表示、この条件リスト、および下記の免責条項を再表示する必要があります。

3. 本ソフトウェアの機能または使用を説明する一切の広告用資料に、以下の確認文を表示する必要があります。

本製品には、Lawrence Berkeley Laboratory の Computer System Engineering Group により開発されたソフトウェアが含まれています。

4. 事前に書面での明確な許可を得ない限り、本ソフトウェアに由来する製品の保証や宣伝のために、大学や研究所の名前を使用することはできません。

本ソフトウェアは理事および貢献者により「現状のまま」提供されており、商品性および特定目的適合性の黙示的保証を含めてこれに限らず、あらゆる明示または黙示の保証も否認します。いかなる理由であれ、契約責任、厳格責任または不法行為（過失その他を含む）のいずれの責任理論によるものであれ、理事または貢献者は、たとえ損害の可能性を知らされていたとしても、本ソフトウェアの使用から生じる直接損害、間接損害、特別損害、懲罰的損害または派生的損害（代替製品またはサービスの調達、使用、データ、利益の損失、または業務の中断を含むがこれらに限定されない）の一切に対し責任を負いません。

本製品には Sony Computer Science Laboratories Inc. により開発されたソフトウェアが含まれています。Copyright © 1997-2003 Sony Computer Science Laboratories Inc. All rights reserved. 以下の条件を満たす場合、修正の有無を問わず、ソースおよびバイナリ形式での再配布および使用が許可されています

1. ソース・コードの再配布には上記の著作権表示、この条件リスト、および下記の免責条項を維持し明記する必要があります。

2. バイナリ形式での再配布には、配布と共に提供される文書および / または資料に上記の著作権表示、この条件リスト、および下記の免責条項を再表示する必要があります。

本ソフトウェアは SONY CSL および貢献者により「現状のまま」提供されており、商品性および特定目的適合性の黙示的保証を含めてこれに限らず、あらゆる明示または黙示の保証も否認します。いかなる理由であれ、契約責任、厳格責任または不法行為（過失その他を含む）のいずれの責任理論によるものであれ、SONY CSL または貢献者は、たとえ損害の可能性を知らされていたとしても、本ソフトウェアの使用から生じる直接損害、間接損害、特別損害、懲罰的損害または派生的損害（代替製品またはサービスの調達、使用、データ、利益の損失、または業務の中断を含むがこれらに限定されない）の一切に対し責任を負いません。

目次

1
ローカルトラフィック管理の概要

ローカルトラフィック管理について 1-1
ローカルトラフィック管理機能の要約 1-1
特定タイプのアプリケーショントラフィックの管理 1-2
パフォーマンスの最適化 1-3
ネットワークセキュリティの強化 1-5
ローカルトラフィック管理設定の概要 1-7
設定ユーティリティの使用 1-7
ローカルトラフィックオブジェクトの設定 1-7
接続とセッションのタイムアウト設定の管理 1-9
はじめに 1-11
ネットワークマップ機能の使用 1-13
フィルタリングメカニズムについて 1-13
ローカルトラフィックオブジェクトのサマリの表示 1-14
ネットワークマップの表示 1-16
アプリケーションテンプレートについて 1-18
アプリケーションテンプレートのメリット 1-18
アプリケーションテンプレートの利用法 1-18
本ガイドについて 1-22
補足情報の参照 1-22
表記規則 1-23
ヘルプおよびテクニカルサポートリソースへのアクセス 1-25

2
バーチャルサーバの設定

バーチャルサーバおよび仮想アドレスの概要 2-1
バーチャルサーバとは 2-1
仮想アドレスとは 2-2
バーチャルサーバの仕組みについて 2-3
バーチャルサーバのタイプについて 2-5
ホストバーチャルサーバ 2-5
ネットワークバーチャルサーバ 2-5
クラスタリングマルチプロセッシングの概要 2-8
CMP が有効になっている場合のシステム動作の違いについて 2-8
CMP の無効化と再有効化 2-9
冗長システムの CMP について 2-10
バーチャルサーバの作成 2-11
バーチャルサーバと仮想アドレスの設定について 2-14
バーチャルサーバの設定 2-14
仮想アドレスの設定 2-20
バーチャルサーバおよび仮想アドレスの管理 2-21
バーチャルサーバの設定の表示または変更 2-21
仮想アドレスの設定の表示または変更 2-23
バーチャルサーバと仮想アドレスのステータスについて 2-24
バーチャルサーバまたは仮想アドレスの有効化 / 無効化 2-25
バーチャルサーバまたは仮想アドレスの削除 2-26

3
ノードの設定

ノードの概要 3-1
ノードの作成と変更 3-2
ノードの設定 3-4
ノードのアドレスの指定 3-5
ノード名の指定 3-5
ヘルスモニタの割り当て 3-5
アベイラビリティ要件の指定 3-7
重み付けの指定 3-7
コネクションリミットの設定 3-7

ノードの管理 3-8
ノードのリストを表示する 3-8
ノードプロパティの表示 3-8
ノードステータスについて 3-9
ノードの有効化または無効化 3-11
ノードの削除 3-11
モニタの関連付けの削除 3-11

4
ロードバランシングプールの設定

ロードバランシングプールの概要 4-1
ロードバランシングプールについて 4-1
ロードバランシングプールの機能 4-1
ロードバランシングプールの作成と変更 4-2
ロードバランシングプールの作成と実装 4-2
ロードバランシングプールの変更 4-3
プールメンバシップの変更 4-3
プールの設定 4-6
プール名の指定 4-7
ヘルスモニタとプールとの関連付け 4-7
アベイラビリティ要件の指定 4-8
SNAT および NAT の許可 4-8
サービスが利用不可になったときの処理の指定 4-9
スローランプタイムの設定 4-9
Type of Service（ToS）レベルの設定 4-9
Quality of Service（QoS）レベルの設定 4-10
ロードバランシング方式の指定 4-10
プライオリティベースのメンバのアクティベーションの指定 4-14
プールメンバの指定 4-15
プールメンバの設定 4-16
プールメンバの重み付けの指定 4-17
プライオリティベースのメンバのアクティベーションの指定 4-17
コネクションリミットの指定 4-17
明示的なモニタ関連付けの選択 4-17
プールメンバに対する明示的なモニタの関連付けの作成 4-18
アベイラビリティ要件の指定 4-19
プールおよびプールメンバの管理 4-20
プールの管理 4-20
プールメンバの管理 4-23
モニタの関連付けの削除 4-27
プールおよびプールメンバの統計情報の表示 4-27

5
プロファイルについて

プロファイルの概要 5-1
プロファイルタイプ 5-1
デフォルトプロファイル 5-3
カスタムプロファイルと親プロファイル 5-4
プロファイルの要約 5-5
プロファイルの作成と変更 5-6
デフォルトプロファイルをそのまま使用する 5-6
デフォルトプロファイルを変更する 5-7
カスタムプロファイルを作成する 5-8
カスタムプロファイルを変更する 5-9
プロファイルの表示と削除 5-11
プロファイルのリストの表示 5-11
プロファイルの削除 5-12
プロファイルの実装 5-12
詳細情報 5-13

6
アプリケーション層トラフィックの管理

アプリケーション層トラフィック管理の概要 6-1
HTTP 標準プロファイルの設定 6-2
HTTP プロファイル設定について 6-2
HTTP 圧縮の設定 6-11
標準的なクライアントとサーバのシナリオにおける圧縮 6-11
BIG-IP を使用した圧縮 6-11
データ圧縮への HTTP プロファイルの使用 6-13
データ圧縮戦略の使い方 6-22
圧縮プロバイダの概要 6-22
圧縮戦略の選択について 6-22
Adaptive 圧縮戦略の概要 6-23
圧縮統計情報の表示 6-26
RAM キャッシュ機能の設定 6-28
RAM キャッシュの基礎 6-28
RAM キャッシュ設定について 6-32
RAM キャッシュへの HTTP プロファイルの使用 6-32
データ圧縮と RAM キャッシュの最適化 6-34
http-acceleration プロファイルの使い方 6-34
http-wan-optimized-compression プロファイルの使い方 6-35
http-lan-optimized-caching プロファイルの使い方 6-36
http-wan-optimized-compression-caching プロファイルの使い方 6-37
FTP プロファイルの設定 6-38
プロファイル名の指定 6-38
親プロファイルの指定 6-38
Translate Extended 値の指定 6-39
データポートの指定 6-39
FTP トラフィックのセキュリティの有効化 6-39
SIP プロファイルの設定 6-40
SIP プロファイル設定の概要 6-40
SIP パーシステンスのカスタマイズ 6-43
RTSP プロファイルの設定 6-44
iSession プロファイルの設定 6-47

7
セッションパーシステンスの有効化

セッションパーシステンスの概要 7-1
パーシステンスプロファイルの設定 7-1
iRules を介したセッションパーシステンスの有効化 7-2
セッションパーシステンスを有効にした OneConnect プロファイルの使い方 7-2
パーシステンスの種類とそれらのプロファイル 7-4
パーシステンスの種類 7-4
セッションパーシステンスの基準について 7-5
Cookie パーシステンス 7-7
宛先アドレスのアフィニティパーシステンス 7-10
ハッシュパーシステンス 7-12
Microsoft Remote Desktop Protocol パーシステンス 7-14
SIP パーシステンス 7-17
送信元アドレスのアフィニティパーシステンス 7-19
SSL パーシステンス 7-20
ユニバーサルパーシステンス 7-21

8
プロトコルプロファイルの管理

プロトコルプロファイルの概要 8-1
Fast L4 プロファイルの設定 8-2
Fast L4 プロファイル設定について 8-2

PVA ハードウェア高速化の設定 8-5
Fast HTTP プロファイルの設定 8-7
Fast HTTP プロファイル設定について 8-8
HTTP Class プロファイルの設定 8-10
HTTP Class プロファイルの設定について 8-10
ほかの BIG-IP モジュールによるトラフィック分類 8-13
TCP プロファイルの設定 8-14
TCP プロファイル設定について 8-14
tcp-lan-optimized プロファイルの設定について 8-19
tcp-wan-optimized プロファイルの設定について 8-20
UDP プロファイルの設定 8-21
SCTP プロファイルの設定 8-22

9
SSL トラフィックの管理
SSL トラフィック管理の概要 9-1
クライアント側およびサーバ側トラフィックの管理 9-1
SSL トラフィック制御機能の要約 9-2
証明書の検証について 9-3
証明書の失効について 9-5
暗号化と復号化について 9-5
クライアント承認について 9-6
SSL セッションパーシステンスについて 9-6
その他の SSL 機能について 9-6
キーおよび証明書の管理 9-7
既存のキーおよび証明書に関する情報の表示 9-7
新規の証明書およびキーに対する要求の作成 9-7
証明書の更新 9-9
証明書とキーのペアの削除 9-10
キー、証明書、およびアーカイブのインポート 9-10
アーカイブの作成 9-11
SSL プロファイルについて 9-11
HTTP プロファイルの設定 9-13
プロファイル名の指定 9-14
親プロファイルの選択 9-14
証明書の名前の指定 9-14
キーの名前の指定 9-15
パスフレーズの指定と確認 9-15
証明書チェーンの設定 9-15
信頼できるクライアント CA の指定 9-15
SSL 暗号の指定 9-16
回避策の設定 9-17
ModSSL 方式のエミュレーションの有効化 9-20
SSL セッションキャッシュの設定 9-20
アラートタイムアウト値の指定 9-21
ハンドシェイクタイムアウト値の指定 9-22
SSL セッションの再ネゴシエーションの強制 9-22
SSL シャットダウンの設定 9-23
非 SSL 接続の受け入れ 9-24
クライアントおよびサーバ認証の設定 9-25
Configuring certificate presentation 9-26
セッションごとの認証の設定 9-27
信頼できるクライアント CA のリストのアドバタイズ 9-28
認証レベルの設定 9-29
名前ベースの認証の設定 9-29
証明書の失効 9-29

10
アプリケーショントラフィックの認証
リモート認証の概要 10-1

BIG-IP 認証モジュール 10-1
認証モジュールの実装 10-2
LDAP 認証モジュールの実装 10-4
LDAP 構成オブジェクトの作成 10-5
LDAP プロファイルの作成 10-7
RADIUS 認証モジュールの実装 10-10
RADIUS サーバオブジェクトの作成 10-10
RADIUS 構成オブジェクトの作成 10-11
RADIUS プロファイルの作成 10-13
TACACS+ 認証モジュールの実装 10-16
TACACS+ 構成オブジェクトの作成 10-16
TACACS+ プロファイルの作成 10-18
SSL クライアント証明書 LDAP 認証モジュールの実装 10-21
SSL クライアント証明書認証とは 10-21
SSL クライアント証明書 LDAP 構成オブジェクトの作成 10-24
SSL クライアント証明書 LDAP 承認プロファイルの作成 10-26
SSL OCSP 認証モジュールの実装 10-29
OCSP の概要 10-29
OCSP レスポンダオブジェクトの作成 10-32
SSL OCSP 構成オブジェクトの作成 10-34
SSL OCSP プロファイルの作成 10-35
CRLDP 認証モジュールの実装 10-38
CRLDP サーバオブジェクトの作成 10-39
CRLDP 構成オブジェクトの作成 10-40
CRLDP プロファイルの作成 10-41
Kerberos 委任認証モジュールの実装 10-44
Kerberos 委任構成オブジェクトの作成 10-45
Kerberos 委任プロファイルの作成 10-46

11
その他のプロファイルの使用
その他のタイプのプロファイルの概要 11-1
OneConnect プロファイルの設定 11-2
NTLM プロファイルの設定 11-4
Statistics プロファイルの設定 11-6
Stream プロファイルの設定 11-7

12
モニタの設定
モニタの概要 12-1
モニタタイプの要約 12-2
モニタの設定について 12-4
モニタ実装の概要 12-4
カスタムモニタの作成 12-8
特殊な設定に関する留意事項 12-10
宛先の設定 12-10
Transparent モードと Reverse モード 12-11
Manual Resume の設定 12-12
Check Until Up の設定 12-14
モニタのプールとノードへの関連付け 12-15
モニタの管理 12-17

13
NAT の設定
ネットワークアドレス変換（NAT）の概要 13-1
NAT とは？ 13-1
受信接続用の NAT について 13-2
発信接続用の NAT について 13-4
NAT の作成 13-6

その他の制約事項 13-7
NAT の管理 13-8
NAT の確認 / 変更 13-8
NAT の削除 13-8
ロードバランシングプールでの NAT の有効化 / 無効化 13-9

14
SNAT の設定

セキュアネットワークアドレス変換の概要 14-1
クライアントで開始された（受信）接続での SNAT の概要 14-2
サーバで開始された（発信）接続での SNAT の概要 14-4
SNAT 実装の概要 14-5
SNAT プールの作成 14-7
SNAT の作成 14-9
標準 SNAT の作成 14-9
インテリジェント SNAT の作成 14-12
SNAT プールのバーチャルサーバへの直接割り当て 14-13
SNAT の管理 14-14
SNAT、SNAT プールの表示または変更 14-14
変換アドレスの定義または表示 14-15
SNAT、SNAT プール、変換アドレスの削除 14-16
ロードバランシングプールでの SNAT または NAT の有効化または無効化 14-17
SNAT 変換アドレスの有効化または無効化 14-17
SNAT の例 14-18
例 1 - SNAT プールを使用する標準 SNAT の確立 14-18
例 2 - インテリジェント SNAT の確立 14-19

15
トラフィッククラスの設定

トラフィッククラスの概要 15-1
トラフィッククラスの作成 15-1
トラフィッククラスの設定 15-2
トラフィッククラスの管理 15-3

16
レートシェーピングの設定

レートシェーピングの概要 16-1
レートクラスの作成と実装 16-2
レートクラス設定の設定 16-4
名前の指定 16-5
ベースレートの指定 16-5
上限の指定 16-5
バーストサイズの指定 16-6
方向の指定 16-8
親クラスの指定 16-8
シェーピングポリシーの指定 16-9
キュー方式の指定 16-9
破棄ポリシーの指定 16-10
レートクラスの管理 16-11

17
iRule の記述

iRule の概要 17-1
iRule について 17-1
基本 iRule 要素 17-2
トラフィックの宛先とアドレス変換の指定 17-4
iRule の作成 17-7

iRules と管理パーティションについて 17-7
iRule 評価の制御 17-9
設定の前提条件 17-9
イベントの指定 17-9
iRule コマンドの使用 17-12
ステートメントコマンド 17-12
クエリおよび操作コマンド 17-12
ユーティリティコマンド 17-13
プロファイルの処理 17-14
プロファイル設定の読み込み 17-14
プロファイル設定のオーバーライド 17-14
iRule でのセッションパーシステンスの有効化 17-15
データグループの作成、管理、使用 17-16
matchclass コマンドの使用 17-16
データグループの作成 17-17
ストレージオプション 17-19
データグループプロパティの表示 17-20
データグループメンバの管理 17-21

A
ヘルスモニタとパフォーマンスモニタ
モニタリファレンス A-1
バーチャルサーバに関する注意事項 A-20
CMP に関する注意事項 A-21
MSSQL の前提条件のタスク A-26
MSSQL モニタの設定とそのデフォルト値 A-26
MSSQL ログインのトラブルシューティング A-28

B
モニタについての追加留意事項
モニタへの Dynamic Ratio ロードバランシングの実装 B-1
Real Server モニタの実装 B-1
WMI モニタの実装 B-3
SNMP DCA および SNMP DCA Base モニタの実装 B-10
MSSQL モニタの実装 B-11

C
OpenSSL ユーティリティの使い方
BIG-IP の OpenSSL について C-1
クライアントシステム証明書の作成 C-1
Web サイト証明書の作成 C-2
証明書失効の使い方 C-3
その他の証明書関連タスクの実行 C-4

用語集

索引

I

ローカルトラフィック管理の概要

- ローカルトラフィック管理について
- ローカルトラフィック管理設定の概要
- ネットワークマップ機能の使用
- アプリケーションテンプレートについて
- 本ガイドについて
- ヘルプおよびテクニカルサポートリソースへのアクセス

ローカルトラフィック管理について

BIG-IP ローカルトラフィック管理システムは、特にローカルネットワークトラフィックの管理用に設計されています。***ローカルトラフィック管理***とは、イントラネットなどのローカルエリアネットワーク（LAN）を出入りするネットワークトラフィックを管理するプロセスのことです。

この設定ガイドは、BIG-IP 製品ファミリに含まれるローカルトラフィック管理製品セットに適用されます。

BIG-IP でよく使用される機能は、ネットワークサーバ上の負荷をインテリジェントに調整するために、着信ネットワークトラフィックをインターセプトおよびリダイレクトする機能です。ただし、サーバの負荷を調整することだけがローカルトラフィック管理ではありません。BIG-IP には、ヘッダやコンテンツデータの検査および変換、SSL 証明書ベースの認証管理、HTTP 応答の圧縮などを実行するさまざまな機能が用意されています。これらの機能を実行するときに、BIG-IP は、トラフィックを適切なサーバのリソースにダイレクトするだけでなく、ネットワークセキュリティを強化します。また、通常は Web サーバで実行されるタスクを実行することで、サーバのリソースを解放します。

◆注

*BIG-IP® LTM は、BIG-IP 製品ファミリを構成する一製品です。BIG-IP 製品ファミリのすべての製品は、TMOS™ とも呼ばれる、強力な Traffic Management Operating System で実行します。すべての BIG-IP 製品サービスの概要については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。BIG-IP システム設定の基本情報については、『**BIG-IP® Systems: Getting Started Guide**』を参照してください。*

ローカルトラフィック管理機能の要約

BIG-IP を適切に設定すると、以下のような多岐にわたるトラフィック管理機能を実行できます。

- トラフィックのバランスを取ることでネットワーク上のサーバの負荷を調整して分散させ、スケーラビリティを実現します。
- HTTP データ圧縮、SSL 認証、SSL 暗号化などの標準的なサーバタスクの負荷を軽減することで、サーバのパフォーマンスを向上させます。
- ネットワーク上のサーバの健全性およびパフォーマンスを監視し、アベイラビリティを実現します。
- セッションと接続のパーシステンス（接続維持）を確立および管理します。
- ユーザ名とパスワードおよび SSL 証明書に基づいたアプリケーショントラフィックの認証および許可機能を処理します。
- パケットスループットを管理して、特定タイプの接続パフォーマンスを最適化します。

- 複数のクライアントリクエストをサーバ側の接続プールに集約することで、パフォーマンスを向上させます。このようなクライアント要求の集約機能は、BIG-IP の OneConnect™ 機能に含まれています。
- 構成設定を適用して、アプリケーション固有のトラフィック（HTTP トラフィック、SSL トラフィックなど）のフローをカスタマイズします。
- 業界標準の Tool Command Language（Tcl）に基づいたユーザ作成スクリプトに従って、特定の接続の管理をカスタマイズします。
- マルチプロセッサプラットフォームでは、BIG-IP の TMM（Traffic Management Microkernel）サービスの複数のインスタンスを使ってトラフィックを処理するように設定することで、トラフィック管理性能を強化しています。

ここで列挙した機能には、ネットワークサーバ上の負荷のバランスを取る基本機能を提供するものと、注目すべき特別な機能を提供するものとがあります。これらの機能には、特定タイプのアプリケーショントラフィックの管理、サーバパフォーマンスの最適化、ネットワークのセキュリティの強化などがあります。続く 3 つのセクションでは、このような特別な機能について説明します。

特定タイプのアプリケーショントラフィックの管理

ローカルトラフィック管理の主要機能として、アプリケーション固有のトラフィックフローをカスタマイズする構成設定を適用することがあります。BIG-IP では、さまざまな種類のトラフィックを、それぞれ異なる方法で制御することが可能です。その場合、各タイプのネットワークトラフィックを管理するためのポリシーを確立します。BIG-IP で管理可能なトラフィックタイプの例は、次のとおりです。TCP、UDP、HTTP、FTP、SSL、Session Initiation Protocol（SIP）。

このような異なるトラフィックタイプを体系的に管理する個別のポリシーを作成することに加えて、次の作業も行えます。

- **iRules™ を作成して、個々のアプリケーション固有の接続に特定の動作を割り当てます。**
  iRules では、特定タイプのトラフィックのコンテンツ（HTTP リクエストまたは応答など）を検索し、それに応じてトラフィックをダイレクトすることが可能です。
- **アプリケーション固有の要求にヘッダデータを挿入します。**
  HTTP リクエストなどのトラフィックにヘッダデータを挿入し、そのヘッダデータに基づき要求の送信先を指定することが可能です。
- **セッションのパーシステンスを実装します。**
  BIG-IP の強力な設定ツールを使用すると、HTTP Cookie、送信元 IP アドレス、宛先 IP アドレス、SSL セッション ID などのデータに基づいて、セッションのパーシステンスを設定できます。
- **ノードまたはプールメンバの健全性やパフォーマンスを監視します。**
  たとえば、BIG-IP ではネットワーク上の Web サーバを監視し、ターゲットの Web サーバが機能していないと判断すると、別のサーバに要求をリダイレクトします。

- **動的比率ロードバランシングアルゴリズムを使用します。**
  このアルゴリズムは特定タイプのサーバ（Windows Management Infrastructure（WMI）サーバなど）の現在の負荷を評価し、その評価に基づき要求をリダイレクトします。特定タイプのアプリケーションに対応してサーバを監視する機能は、ネットワークの最適なパフォーマンスを維持するための重要な手段となります。

最後に、BIG-IP では一連のアプリケーションテンプレートを利用できるようになっています。テンプレートではウィザードに似た画面が表示され、特定アプリケーションのトラフィックを処理するようにシステムを設定することができます。BIG-IP が提供するテンプレートには、BEA WebLogic や Microsoft® IIS などがあります。

パフォーマンスの最適化

BIG-IP には、サーバパフォーマンスを最適化するための機能が複数用意されています。これらの機能は、労働集約型のトラフィック管理タスク（SSL 証明書の検証など）の負荷を軽減するか、またはサーバ側接続のプーリング、再利用、および全体的なパーシステンスを有効にします。

サーバタスクの負荷の軽減

BIG-IP では、ネットワークサーバから以下のタスクの負荷をオフロードすることができます。

- 証明書失効リスト（CRL）、Online Certificate Status Protocol（OCSP）、Certificate Revocation List Distribution Point（CRLDP）技術のいずれかを使った証明書の失効状態のチェックを含む、SSL 証明書ベースの認証
- SSL 暗号化および復号化
- リモート LDAP サーバを使用する SSL 証明書ベースの許可
- HTTP データ圧縮と RAM キャッシュ
- LDAP および RADIUS サーバなど、リモート認証サーバを使用した場合のアプリケーショントラフィックの認証および承認
- Microsoft® Remote Desktop 接続の書き換え

TCP 接続および HTTP 接続の最適化

BIG-IP では、TCP 接続と HTTP 接続を特定の方法で管理して、サーバパフォーマンスを最適化します。主なネットワーク最適化機能には、OneConnect™、HTTP パイプライニング、HTTP データ圧縮、RAM キャッシュ、レートシェーピングなどがあります。

OneConnect

OneConnect™ 機能には、次のコンポーネントが含まれます。

- **コンテンツスイッチング**
  HTTP クライアントが単一の接続で複数の要求を送信する場合、BIG-IP ではこれらの要求を個別に処理し、必要に応じて異なる宛先サーバに送信することが可能です。
- **コネクションプーリング**
  この機能により BIG-IP は、開いてはいるが使用されてないサーバ側の接続を結合し、ほかのクライアントで使用できるようにします。これにより、クライアントリクエストを処理するために必要なサーバの数を大幅に減らすことができます。
- **OneConnect 変換**
  HTTP/1.0 要求に対して **Keep-Alive** サポートを HTTP **Connection** ヘッダに追加することで、サーバ側の接続を確実に開いたままにする場合があります。HTTP **Connection** ヘッダをこのように操作する機能は、OneConnect 変換と呼ばれています。この機能はコネクションプーリングと連動します。

OneConnect 機能の一般情報については、第 6 章「*アプリケーション層トラフィックの管理*」および第 11 章「*その他のプロファイルの使用*」を参照してください。

HTTP パイプライニング

OneConnect 機能に加えて、BIG-IP には、パイプライン化された要求を処理する機能があります。つまり BIG-IP では、前の要求の応答を待たずに、複数のクライアントリクエストを発行できるようになっています。パイプライニングは、HTTP/1.1 要求のみに使用可能な最適化機能です。

HTTP パイプライニングの詳細については、第 6 章「*アプリケーション層トラフィックの管理*」を参照してください。

HTTP データ圧縮

HTTP 応答を圧縮するように HTTP プロファイルを設定すると、バックエンドサーバの負荷を緩和することができます。BIG-IP が HTTP 応答を圧縮すれば、HTTP トラフィックを処理するバックエンドサーバが、データ圧縮にリソースを使用する必要がありません。

広域ネットワーク（WAN）のいずれかの端部に Local Traffic Manager システムが設置されていれば、iSession 設定プロファイルを設定することができます。これにより最適化トンネルが作成され、BIG-IP から WAN 経由でもう一方の端部に送られるデータが圧縮されることになります。

RAM キャッシュ

BIG-IP の RAM に HTTP オブジェクトを格納できます。その後の接続では、これらのオブジェクトを再利用し、バックエンドサーバの負荷を緩和することができます。

レートシェーピング

*レートシェーピング*は、特定タイプの接続のスループットをカスタマイズするために、接続をレートクラスに分類できる機能です。この機能は、優先インターネットカスタマ用に Web サーバのパフォーマンスを最適化する必要がある場合などに便利です。

TCP 最適化

BIG-IP には、順序どおりの配信、コンテンツスプーリングなど、重要な TCP 最適化機能が含まれています。

ネットワークセキュリティの強化

セキュリティは、ローカルネットワークトラフィックを管理する上で重要な考慮事項です。そのため、BIG-IP には、セキュリティ侵犯の防止を考慮して設計された機能が数多く用意されています。これらの機能は、ユーザやアプリケーションの認証および許可だけでなく、侵入検知や DoS 攻撃の緩和にも関連しています。

一般に、BIG-IP によってセキュリティ問題が検出された場合、以下のような処理が行われます。

- SSL 証明書の検証に基づいてクライアントリクエストを拒否する
- 許可されていないパケットを拒否および廃棄する
- システム管理者に攻撃または侵入に対する警報を出す
- 不審なトラフィックを特定のターゲットサーバにダイレクトする
- 認証の失敗をログに記録する
- SYN フラッド攻撃を防止する

あらゆるネットワーク環境で重要な考慮事項は、ユーザとそのクライアントリクエストを認証するため、またサーバリソースへのユーザおよびアプリケーションアクセスを制御するために使用する認証および許可メカニズムです。このため、BIG-IP では、Pluggable Authentication Module（PAM）テクノロジをサポートしています。また、認証や許可のニーズに対応するために選択可能な PAM 認証モジュールの完全なセットを提供しています。

BIG-IP に用意されている認証モジュールは、次のとおりです。

- **LDAP モジュール**
  リモート LDAP サーバを使用して、ユーザ名とパスワードのユーザ認証を実行します。
- **RADIUS モジュール**
  Remote Authentication Dial In User Service（RADIUS）サーバを使用して、ユーザ名とパスワードのユーザ認証を実行します。
- **TACACS+ モジュール**
  リモート Terminal Access Controller Access Control System（TACACS+）サーバを使用して、ユーザ名とパスワードのユーザ認証を実行します。
- **SSL クライアント証明書 LDAP モジュール**
  リモート LDAP サーバを使用して、クライアント SSL トラフィックの SSL 証明書ベースの許可を実行します。

- **OCSP モジュール**
  リモート Online Certificate Status Protocol（OCSP）サーバを使用して、クライアントおよびサーバの SSL トラフィックを認証するために、最新の SSL 証明書の失効ステータスを提供します。
- **CRLDP モジュール**
  クライアントおよびサーバのSSLトラフィックを認証するために、業界標準の Certificate Revocation List Distribution Point（CRLD）技術を利用して、SSL 証明書の失効状態を管理します。
- **Kerberos 委任モジュール**
  Microsoft Windows Integrated Authentication を使ってクライアントトラフィックを認証します。

ローカルトラフィック管理設定の概要

基本ネットワークを設定しており、BIG-IP への管理アクセス権を持ち、インターフェイスごとに少なくともデフォルトの VLAN 割り当てがある場合、次の手順として、内部サーバをターゲットとするトラフィックの管理用ネットワークを設定します。詳細については、『***BIG-IP® Systems: Getting Started Guide***』を参照してください。

BIG-IP の中心には、バーチャルサーバとロードバランシングプールがあります。バーチャルサーバは、着信トラフィックを受信し、基本的な送信元 IP アドレスおよび宛先 IP アドレス変換を実行し、ロードバランシングプール内で同じグループに属しているサーバにトラフィックをダイレクトします。

設定ユーティリティの使用

すべてのユーザは、BIG-IP を初めて使用するときに、ライセンスを取得して設定するために Web ベースの設定ユーティリティを使用する必要があります。

セットアップユーティリティでの管理ネットワークの設定とトラフィック管理ソフトウェアの初期設定に加え、設定ユーティリティを使ってBIG-IPのカスタマイズと保守を行います。設定ユーティリティでは、ほかにも現在のシステムパフォーマンスを監視したり、ネットワークマップ（ユーザが作成したバーチャルサーバが、それらのバーチャルサーバが参照するプール（およびプールメンバ）と共に示されます）を確認したりすることができます。

設定ユーティリティのユーザプリファレンス設定の詳細については『***TMOS® Management Guide for BIG-IP® Systems***』を、サポートされるブラウザの詳細については、F5 Prime Members Web サイトのリリースノートを参照してください。

ローカルトラフィックオブジェクトの設定

基本的なローカルトラフィック管理システムを設定するには、設定ユーティリティを使用します。このユーティリティを使用すると、構成オブジェクトの完全なセットを作成できます。これはローカルトラフィック管理を実行する上で連携して機能します。各オブジェクトには構成設定のセットが含まれています。このセットをそのまま使用することもできますが、必要に応じて変更することもできます。

ローカルトラフィックオブジェクトを設定するには、オブジェクトを 1 種類ずつ設定していきますが、場合によっては*アプリケーションテンプレート*機能を使って必要なオブジェクトをまとめて設定することもできます アプリケーションテンプレートの使用法については、「*アプリケーションテンプレートについて*」（1-18 ページ）を参照してください。

以下のローカルトラフィックオブジェクトを設定することができます。

- **バーチャルサーバ**
  バーチャルサーバの主要な機能は、要求を受信し、指定した基準に基づいてそれをプールメンバに分散することです。

- **アプリケーションプロファイル**
  アプリケーションタイププロファイルには、FTP や HTTP などのプロトコルを使用するさまざまなアプリケーションタイプの動作を定義する設定が含まれます。
- **セッションのパーシステンスプロファイル**
  セッションのパーシステンスプロファイルを使用すると、さまざまな基準（HTTP Cookie、送信元 IP アドレス、宛先 IP アドレスなど）に基づいてセッションのパーシステンスを実装できます。
- **プロトコルプロファイル**
  プロトコルプロファイルには、レイヤ 4 トラフィック（TCP、UDP、SCTP トラフィックなど）の動作を定義する設定が含まれます。
- **SSL プロファイル**
  SSL プロファイルによって、BIG-IP 経由で送信されるクライアント側とサーバ側の両方のデータの暗号化と復号化が可能になります。
- **認証プロファイル**
  認証タイププロファイルは、リモート認証サーバと BIG-IP の間の認証の互換性を提供します。
- **iRules**
  iRules は、プールメンバの選択基準を定義し、コンテンツ変換、ロギング、カスタムプロトコルサポートなどを実行が可能です。iRules の詳細については、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。
- **ロードバランシングプール**
  ロードバランシングプールには、処理対象の要求を送信できるサーバが含まれます。
- **ノード**
  ノードは、ネットワーク上で有効または無効にすることができ、ステータスを取得できるサーバ IP アドレスを表します。
- **モニタ**
  モニタは、プールメンバの現在の健全性またはパフォーマンスを追跡します。
- **レートクラス**
  レートクラスは、帯域幅消費の制御に使用するレートシェイピングを実行します。
- **トラフィッククラス**
  トラフィッククラスは、ユーザ定義の一連の基準に基づきトラフィックフローの調整を行います。
- **SNAT**
  SNAT（Secure Network Address Translation）は、要求の送信元 IP アドレスを変換し、複数のホストが同じ送信元 IP アドレスを共有できるようにします。
- **SSL 証明書**
  SSL 証明書オブジェクトを使用すると、SSL 接続の終端および開始を目的として、BIG-IP に SSL 証明書要求を生成したり、SSL 証明書をインストールすることができます。

BIG-IP でローカルトラフィック管理用に設定する必要があるもっとも一般的なオブジェクトは、以下の 3 つとなっています。

- バーチャルサーバ
- ロードバランシングプール
- プロファイル

オブジェクトを 1 つずつ設定するか、もしくはアプリケーションテンプレートを利用して、システムのローカルトラフィックオブジェクトを設定し終えたら、ネットワークマップを使ってそれらのオブジェクトを表示することができます。***ネットワークマップ***には、ユーザが行ったローカルトラフィック設定を視覚化したもの（バーチャルサーバやそれらの関連プールなど）のほか、既存のバーチャルサーバ、プール、ノード、iRules™ に関する統計情報もあわせて表示されます。ネットワークマップの詳細については、「*ネットワークマップ機能の使用*」（1-13 ページ）を参照してください。

接続とセッションのタイムアウト設定の管理

BIG-IP では、複数のタイムアウトを設定してアクティブ接続管理を簡易化できるようになっています。接続をアクティブな状態のまま維持しつつ、接続テーブルでその接続を追跡管理することで、負荷分散された接続を明示的に管理します。***接続テーブル***には、クライアント側とサーバ側の接続の状態情報、ならびにクライアントとサーバの関係が含まれます。

接続テーブルの接続は、それぞれがシステムリソースを消費して、表のエントリの管理と接続状態の監視を行います。BIG-IP は、貴重なシステムリソースを使いはたしてしまわないように、接続がアクティブでなくなるタイミングを見極め、接続をリタイアさせる必要があります。接続テーブルのデータ量が増えて、チェックが行きとどかなくなると、メモリやプロセッササイクルなどのリソースが不足する危険性があります。

セッションパーシステンスを使用している場合は、パーシステンステーブルのエントリの有効期限もあわせて管理することができます。

接続のリープ

終了やリセットが通常どおり行われた接続については、接続テーブルから自動的に削除されます。しかし、大量の接続の場合、いくつかの理由によって、正常に終了せずにアイドル状態が続くことがよくあります。したがって、接続がアクティブでないと特定された時点で、BIG-IP はそれらをリープする必要があります。***リーピング***とは、そのままではアイドル状態が続くことになる接続をリタイアまたはリサイクルさせるプロセスです。

積極的なリーピングを促進するために、いくつかのタイムアウト値を設定して、指定された時間を過ぎてアクティブではなくなったと思われる接続を切断することができます。これらのタイムアウト設定の中にはユーザが設定できないものも少数含まれるものの、ほとんどのタイムアウト設定については、あらゆるアプリケーションのニーズに応じて積極的に設定することが可能です。

タイムアウト 設定は複数の場所で設定できるため、同じ 1 つの接続に複数のタイムアウト 設定が影響をおよぼす場合があることを知っておくことが重要です。最適なタイムアウト 設定とは、システムリソースを

保護するために、適当な時間（アプリケーションによって異なります）アイドル状態が続いた後で、その接続が非アクティブでありリタイアが必要だということを決定するというものです。

アイドルタイムアウトオプション

アイドル接続は、接続を処理するバーチャルサーバに関連付けられたプロトコルプロファイルまたは SNAT によってタイムアウトさせることができます。バーチャルサーバが管理していない接続については、SNAT 自動マップ設定もしくは VLAN グループ設定に基づくタイムアウトが可能です。

表 1.1 は、リーピングに影響を与えるアイドル接続のタイムアウト設定を含むオブジェクトの一覧となっています。オブジェクトの種類ごとに、デフォルト値と、その値がユーザによる設定に対応しているかどうかを示しています。

設定オブジェクトのタイプ	デフォルト値（秒）	ユーザによる設定
Fast L4、Fast HTTP、TCP、SCTP プロファイル	300	可
UDP プロファイル	60	可
SNAT 自動マップ	300	×
VLAN グループ	300	×

***表 1.1** 接続のリープに影響するアイドルタイムアウト設定*

接続に適用されるもっとも短いタイムアウト値が、常に有効な値ということになります。しかし場合によっては、この挙動を変更したほうがいいケースもあります。

たとえば、長時間の接続を処理するフォワーディングバーチャルサーバを設定している場合、それらの接続はアイドル状態が長く続くことになります（SSH セッションなど）。このような場合、関連するプロトコルプロファイル（この場合は TCP）に対して長めのアイドルタイムアウト値を設定することが可能です。

しかし、SNAT 自動マップ機能も有効になっている場合は、300 秒というデフォルトの固定タイムアウト値がそのまま有効となります。

その他のタイムアウト設定

BIG-IP にはほかに 2 つのアイドルタイムアウト設定が含まれますが、これらが接続のリープに影響を与えることはありません。これらの設定が含まれるのは、OneConnect とパーシステンスの 2 つのプロファイルタイプです。表 1.2 では、これらの設定のデフォルト値と、ユーザによる設定が可能な値かどうかを示しています。

設定オブジェクトのタイプ	デフォルト値（秒）	ユーザによる設定
OneConnect プロファイル	無効	可
Cookie Hash、宛先アドレスアフィニティ、Hash、SIP、送信元アドレスアフィニティ、ユニバーサルパーシステンスプロファイル	180	可
MSRDP および SSL パーシステンスプロファイル	300	可

***表 1.2** 接続のリープに影響しないアイドルタイムアウト設定*

OneConnect タイムアウト値は、アイドル状態のサーバ側の接続を再利用のために利用できる時間を制御します。つまり、このタイムアウト値次第では、サーバ側の接続が一定時間アイドル状態になった後に閉じられてしまう場合があるということです。このような場合、その接続はアクティブに使用されていなかったため、クライアント側のアクティブな接続が影響を受けることはなく、別のサーバ側の接続が新しい接続として透過的に選択もしくは確立されることになります。OneConnect タイムアウト設定は、ほかのプロファイルのアイドルタイムアウト設定に合わせる必要はありません。

パーシステンスタイムアウト設定は、接続ではなくセッションのアイドルタイムアウトを指定します。したがって、たとえセッション内の接続が切断された場合でもセッションを継続できるように、パーシステンスタイムアウト値には適用可能な接続アイドルタイムアウト値よりわずかに大きな値を指定するようにします。

はじめに

有効なトラフィック管理設定システムを設定するには、バーチャルサーバとロードバランシングプール、そして場合によってはプロファイルを作成する必要があります。

バーチャルサーバの設定

バーチャルサーバを作成する際は、バーチャルサーバのタイプ（ホストバーチャルサーバまたはネットワークバーチャルサーバ）を指定します。その後、さまざまなプロパティおよびリソース（アプリケーション固有のプロファイル、セッションのパーシステンス、プール選択基準を定義する iRules と呼ばれるユーザ作成スクリプトなど）を付加できます。これらのプロパティおよびリソースがすべてバーチャルサーバに関連付けられることで、BIG-IP によるローカルトラフィックの管理方法が決定します。バーチャルサーバの作成と設定には、設定ユーティリティの [Virtual Servers] 画面を使用します。

バーチャルサーバの詳細については、第 2 章「*バーチャルサーバの設定*」を参照してください。

ロードバランシングプールの設定

*ロードバランシングプール*は、クライアントリクエストを処理するために同じグループにまとめられた内部サーバの集合です。プール内のサーバは*プールメンバ*と呼ばれます。BIG-IP では、Round Robin と呼ばれるデフォルトのロードバランシングアルゴリズムを使用して、クライアントリクエストをそのプールのメンバに送信します。

ロードバランシングプールを実装するには、まずプールを作成し、次にプール名を既存のバーチャルサーバに関連付けます。バーチャルサーバは、関連付けられている 1 つまたは複数のプールにクライアントリクエストを送信します。

プールには、プールメンバの IP アドレス、ロードバランシングモード、ヘルスモニタおよびパフォーマンスモニタなど、関連付けられた設定があります。プールを作成する際は、これらの設定のいくつかにデフォルト値を使用することも、必要に応じてデフォルト値を変更することもできます。ロードバランシングプールを作成して設定するには、設定ユーティリティの [Pools] 画面を使用します。

ロードバランシングプールの詳細については、第 4 章「*ロードバランシングプールの設定*」を参照してください。

プロファイルの設定

*プロファイル*は、HTTP 接続など、特定タイプのネットワークトラフィックに適用される構成設定のグループです。バーチャルサーバで特定タイプのトラフィックを管理する場合、適切なプロファイルをバーチャルサーバに関連付けると、バーチャルサーバによってそのプロファイルの設定が該当タイプのすべてのトラフィックに適用されます。

たとえば、BIG-IP で HTTP 応答データを圧縮する場合、圧縮が有効になるように HTTP プロファイルを設定し、そのプロファイルをバーチャルサーバに関連付けることができます。その後、バーチャルサーバが HTTP リクエストを処理するとき、BIG-IP によって応答が圧縮されます。

各自のニーズに合わせて作成できるプロファイルのタイプには、HTTP、FTP、SIP、RTSP、Persistence、Fast L4、Fast HTTP、HTTP Class、TCP、UDP、SCTP、Client and Server SSL、Authentication、OneConnect、NTLM、Statistics、Stream などがあります。プロファイルを作成する際は、設定にデフォルト値を使用することも、必要に応じてデフォルト値を変更することもできます。プロファイルの作成と設定には、設定ユーティリティの [profiles] 画面のいずれか 1 つを使用します。

プロファイルの設定の詳細については、第 5 章「*プロファイルについて*」および次のいずれかの章を参照してください。

- 第 6 章「*アプリケーション層トラフィックの管理*」
- 第 7 章「*セッションパーシステンスの有効化*」
- 第 8 章「*プロトコルプロファイルの管理*」
- 第 9 章「*SSL トラフィックの管理*」
- 第 10 章「*アプリケーショントラフィックの認証*」
- 第 11 章「*その他のプロファイルの使用*」

ネットワークマップ機能の使用

設定ユーティリティには「ネットワークマップ」という機能が含まれます。***ネットワークマップ機能***では、ローカルトラフィックオブジェクトのサマリに加えて、BIG-IP 上のバーチャルサーバ、プール、プールメンバからなる仮想マップが表示されます。ローカルトラフィックサマリとネットワークマップのいずれについても、ユーザ指定の基準に基づき、表示させたい情報をフィルタリングする検索メカニズムを使って、ディスプレイをカスタマイズすることが可能です。検索で一致した箇所は青色で表示されます。

フィルタリングメカニズムについて

ネットワークマップ機能で得られた結果を、フィルタバーの **[Type]** リストと **[Status]** リスト、ならびに **[Search]** ボックスを使ってフィルタリングすることができます。**[Search]** ボックスでは、具体的な文字列を入力することができます（オプション）。図 1.1 は、ネットワークマップ画面のフィルタリングオプションを示しています。

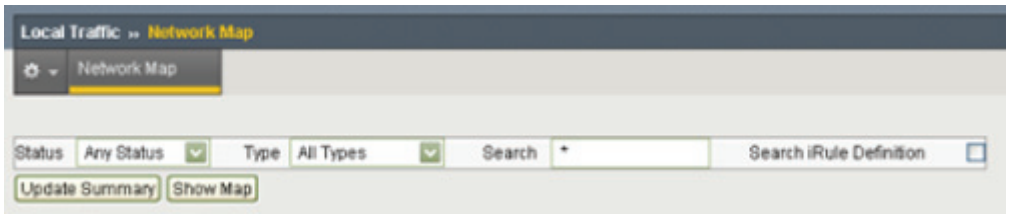

図 1.1 *ネットワークマップ画面のフィルタリングオプション*

[Search] ボックスを使う場合は、検索で使用するテキスト文字列を指定することができます。デフォルトはアスタリスク (*****) です。**[Status]** フィールドと **[Type]** フィールドで、検索の範囲を指定します。システムは指定された検索文字列を使い、画面に表示された結果のフィルタリングを行います。

たとえば、IP アドレスに **10.10** が含まれる無効なノードのみを検索しようとした場合、それらのノードと共に、プールのメンバー、プール、関連するバーチャルサーバ、およびそのバーチャルサーバに明示的に適用された iRule が返されます。結果はバーチャルサーバ名ごとにアルファベット順にソートされます。

BIG-IP では、IPv4 と IPv6 の両方のアドレス形式で、名前、IP アドレス、および IP アドレスとポートの組み合わせで検索を行うことができます。文字列の検索は、アスタリスクワイルドカード文字でその文字列を囲んだ場合と同じように実施されます。たとえば、「**10**」と指定すると、「***10***」と入力したのと同じ検索結果が得られます。また、アスタリスクワイルドカード文字を具体的に指定することも可能です。たとえば、「**10.10.10.*:80**」、「**10.10***」、「***:80**」のような検索文字列を使用することができます。ワイルドカード文字を具体的に指定した場合、それに従い文字列の検索が行われます。たとえば、「**10***」と指定した場合、システムは先頭に「**10**」がつく IP アドレスを持つオブジェクトを検索すると判断します。

◆ヒント

ブラウザが低速になり処理を停止するまでにレンダリングできるデータ量には制限があります。大規模な設定をマッピングした場合これらの制限に達する可能性があることから、メモリ不足によって設定全体のネットワークマップを作成できないことが考えられます。そのような場合、ネットワークマップサマリ画面を使って設定の複雑度を指定できることを告げる警告が表示され、作成されるマップのサイズを知る目安となります。返される結果数が少なくなるように検索基準を変更して、これらの制限に触れることのないマップを作成することができます。

ローカルトラフィックオブジェクトのサマリの表示

初めてネットワークマップ画面を開く際には、ローカルトラフィックオブジェクトのサマリが表示されます。このサマリには、検索メカニズムにより指定されたオブジェクトのタイプ、各種オブジェクトの数、および各種オブジェクトで任意の状態にあるオブジェクトの数が含まれます。

サマリには以下のオブジェクトのデータが表示されます。

- バーチャルサーバ
- Pools
- プールメンバ
- ノード
- iRules

◆注

ローカルトラフィックサマリに含まれるのは、バーチャルサーバにより参照されるオブジェクトに限られます。たとえば、プールを設定したものの、そのプールを参照するバーチャルサーバが存在しないという場合には、ローカルトラフィックサマリにそのプールや、プールメンバー、関連ノードが含まれることはありません。

図 1.2 (1-15 ページ) は、システム上のローカルトラフィックオブジェクトについてまとめたネットワークマップ画面の表示例です。

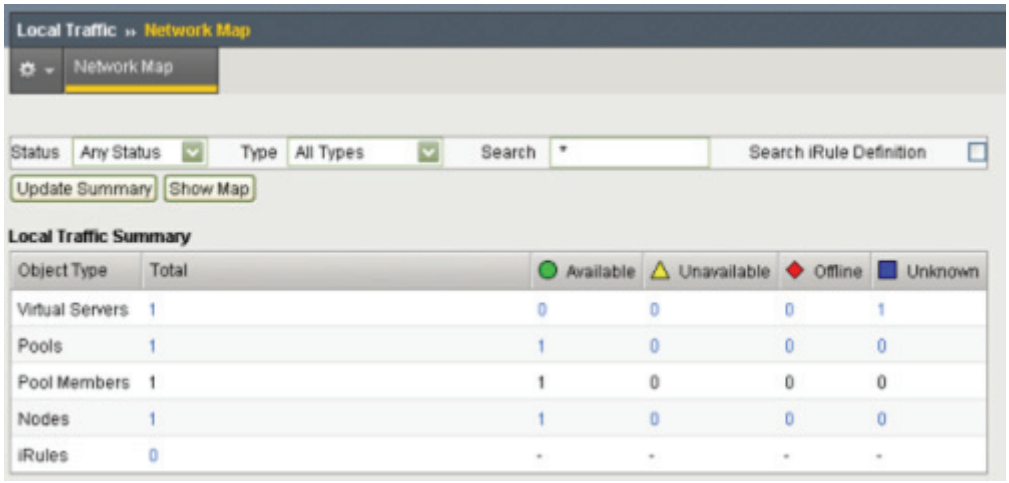

図 1.2 ローカルトラフィックサマリの表示例

サマリにまとめられたオブジェクトの種類ごとに、各状態にあるオブジェクトの数が表示されます。表 1.3 では、サマリ画面でローカルトラフィックオブジェクトに対して表示される状態インジケータについて解説しています。

ステータスインジケータ	説明
	オブジェクトは有効で、利用可能な状態です（トラフィックを受信可能）。
	オブジェクトは有効ですが、現在利用することはできません。ただし、ユーザがアクションを起こさなくても、後で利用可能になる場合があります。 この状態のオブジェクトとしては、コネクションリミットを超えたバーチャルサーバなどがあげられます。接続数が制限値を下回ると、バーチャルサーバは再び利用可能になります。
	オブジェクトは有効ですが、関連付けられたオブジェクトによって「unavailable」とマーク付けされているため、オフラインになっています。この状態を変えて、オブジェクトがトラフィックを受信できるようにするには、ユーザがそのオブジェクトを有効にする必要があります。
	オブジェクトの状態は不明です。

表 1.3 ローカルトラフィックサマリの状態アイコン

ローカルトラフィックオブジェクトのサマリを表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開し、[**Network Map**] をクリックします。
   ネットワークマップ画面が開きます。

2. [**Status**] リストから状態を選択します。
これで、検索を行うとその状態のオブジェクトのみが返されることになります。指定できる値は、**Any Status**（デフォルト値）、**Available**、**Unavailable**、**Offline**、**Unknown** です。
3. [**Type**] リストからオブジェクトの種類を選択します。
これで、検索を行うとその種類のオブジェクトのみが返されることになります。指定できる値は、**All Types**（デフォルト値）、**Virtual Servers**、**iRules**、**Pools**、**Pool Members**、**Nodes** です。
4. 検索結果をさらに絞り込みたい場合は、**Search** ボックスに文字列を入力します。
たとえば、名前に **10.10** という文字列が含まれるオブジェクトのみを検索することができます。

   注：*パフォーマンス上の理由から、通常、iRule テキスト内において特定の検索文字列の検索が行われることはありません。特定の文字列を含む iRule もあわせて検索したい場合は、ステップ 5 を参照してください。そうでない場合はステップ 6 に進んでください。*
5. すべての iRule テキストを検索するには、[**Search iRule Definition**] ボックスをオンにします。

   注：*この設定を有効にすると、検索処理実行時のシステムパフォーマンスに影響が出る可能性があります。*
6. [**Update Summary**] ボタンをクリックします。
画面に表示されたローカルトラフィックサマリがリフレッシュされます。

ネットワークマップの表示

ネットワークマップとは、システムで定義されたバーチャルサーバ、プール、プールメンバ、ノード、iRule の名前と状態を視覚的に階層化したものです。同マップでは、バーチャルサーバを筆頭に、すべてのオブジェクトが文脈に沿って表示されます。画面上端の [**Status**]、[**Type**]、[**Search**] 設定で、マップに含まれるオブジェクトを指定することができます。

マップ上のオブジェクトにカーソルを合わせると、そのオブジェクトの情報が含まれるテキストが表示されます。オブジェクトに付いている状態アイコンにカーソルを合わせると、そのオブジェクトの状態に関する情報が含まれるテキストが表示されます。プールのプロパティ画面でも、これと同じテキストが表示されます。

オブジェクトはアルファベット順に整理され、従属関係が階層的に示されます。

ネットワークマップでの文脈に沿ったオブジェクトの表示法には決まりがあり、最新の画面では、それらのオブジェクトに関連するほかの状態、種類、名前のオブジェクトも一緒に示されるようになっています。なぜならネットワークマップでは、オブジェクトを文脈に沿って表示する際に、かならずそれらに依存するオブジェクトとそれらが依存するオブジェクトと共に表示するためです。

たとえば、利用可能な状態のバーチャルサーバ 1 台に、利用可能な状態のプールが 1 つと、利用可能な状態とオフライン状態のプールメンバが 1 つずつある場合、[**Status**] リストから [**Offline**] を選択すると、オフラインのプールメンバが、利用可能なバーチャルサーバと利用可能なプールと共に、文脈に沿って表示されます。これは、利用可能なバーチャルサーバと利用可能なプールがオフラインのプールメンバに依存しているためです。

ネットワークマップを表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開し、[**Network Map**] をクリックします。
   ネットワークマップ画面が開きます。
2. [**Status**] リストから状態を選択します。
   これで、検索を行うとその状態のオブジェクトのみが返されることになります。指定できる値は、**Any Status**（デフォルト値）、**Available**、**Unavailable**、**Offline**、**Unknown** です。
3. [**Type**] リストからオブジェクトの種類を選択します。
   これで、検索を行うとその種類のオブジェクトのみが返されることになります。
4. 検索結果をさらに絞り込みたい場合は、[**Search**] ボックスに文字列を入力します。
   たとえば、名前に **10.10** という文字列が含まれるオブジェクトのみを検索することができます。

   注：*パフォーマンス上の理由から、通常、iRule テキスト内において特定の検索文字列の検索が行われることはありません。特定の文字列を含む iRule もあわせて検索したい場合は、ステップ 5 を参照してください。そうでない場合はステップ 6 に進んでください。*
5. すべての iRule テキストを検索するには、[**Search iRule Definition**] ボックスをオンにします。

   注：*この設定を有効にすると、検索処理実行時のシステムパフォーマンスに影響が出る可能性があります。*
6. [**Show Map**] ボタンをクリックします。
   要求したネットワークマップが表示されます。

アプリケーションテンプレートについて

*アプリケーションテンプレート*は、ユーザに特定アプリケーションの関連情報を問い合わせる、ウィザードに似た機能です。それらの情報をもとに、BIG-IP でそのアプリケーションのトラフィックを処理するように、自動で設定されるようになっています。

アプリケーションテンプレートには、**Microsoft IIS** テンプレートなどがあります。このテンプレートは、Microsoft IIS サーバに送信されるトラフィックを管理するための BIG-IP オブジェクトを作成するためのものです。

以下のような複数のアプリケーションテンプレートを使用することができます。

- **BEA WebLogic**
- Microsoft Exchange Outlook Web Access
- Microsoft IIS
- Microsoft SharePoint
- Oracle Application Server
- SAP ERP Central Component
- SAP Enterprise Portal
- VMware VDI

アプリケーションテンプレートのメリット

アプリケーションテンプレートを使えば、BIG-IP の設定が簡単に行えます。BIG-IP に管理させたいアプリケーショントラフィックタイプに関連するオブジェクトを 1 つずつ作成する代わりに、アプリケーションテンプレートを実行することができます。アプリケーションテンプレートは、そのアプリケーション専用にカスタマイズされた BIG-IP オブジェクトを自動で作成します。使用しているテンプレートによって、ローカルトラフィックオブジェクト、TMOS オブジェクト、もしくはそれらの両方が作成されることになります。

たとえば、**Microsoft IIS** アプリケーションテンプレートを実行した場合、ユーザが提供する情報を使って、バーチャルサーバ、プール、HTTP プロファイル、ヘルスモニタが作成され、これらはすべて IIS サーバに送信される Web トラフィックを制御するようにカスタマイズされたものとなります。

アプリケーションテンプレートの利用法

アプリケーションテンプレートは設定ユーティリティから実行することができます。

◆ 重要

*アプリケーションテンプレートによって作成されるローカルトラフィックオブジェクトは、すべて **Common** 管理パーティションに置かれます。そのため、アプリケーションテンプレート機能（テンプ*

*レートリスト画面の表示を含む）を使用するには、**Common** パーティションのオブジェクトを表示・管理するためのユーザロールがユーザアカウントに割り当てられている必要があります。*

利用可能なすべてのテンプレートを確認するには、テンプレートリスト画面を表示します。図 1.3 は、設定ユーティリティのテンプレートリスト画面の表示例です。

***図 1.3** テンプレートリスト画面の表示例*

アプリケーションテンプレートについての後半では、アプリ ケーション
テンプレート機能の一般情報を紹介しています。アプリケーションに特化した導入情報を見るには、**www.f5.com** にアクセスして、メニューバーの **[Solutions]** をクリックしてください。ここで以下の導入ガイドを入手することができます。

- ***Deploying the BIG-IP System with BEA WebLogic Server***
- ***Deploying the BIG-IP System with Microsoft Internet Information Services (IIS) 7.0***
- ***Deploying F5 with Microsoft Office SharePoint 2007***
- ***Deploying F5 with Oracle Application Server 10g***
- ***Deploying F5 with SAP NetWeaver and Enterprise SOA***
- ***Deploying F5 with VMware Virtual Desktop Infrastructure (VDI)***

アプリケーションテンプレートの使用法

アプリケーションテンプレートを使用するには、現在の管理パーティションを **[Common]** に設定して、設定ユーティリティからテンプレートを実行します。

利用可能なすべてのテンプレートがテンプレートリストに表示されます（図 1.3 を参照）。次に、テンプレートの名前をクリックして、結果画面に情報を表示させます。

アプリケーションテンプレートを使用するには

1. 現在の管理パーティションが **[Common]** に設定されていることを確認します。

2. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して、[**Templates and Wizards**] をクリックします。
テンプレート画面が開き、テンプレートの一覧が表示されます。
3. [Application] カラムで、テンプレート名をクリックします。
そのアプリケーションの画面が開きます。
4. 画面上のすべての設定項目を設定します。
5. [**Finished**] をクリックします。
そのテンプレートにより作成されたオブジェクトの一覧が表示されます。

Microsoft IIS を初めとするあらゆるアプリケーションテンプレートで、そのテンプレートによる環境を複数作成することができます。たとえば、Microsoft IIS テンプレートを使って、3 つのバーチャルサーバ用に **my_iis1**、**my_iis2**、**my_iis3** という 3 つの環境を個別に作成することが可能です。

HTTP プロファイルの共有

運用の効率化を図るために、BIG-IP では、1 つのアプリケーションテンプレートによる複数の環境で、そのテンプレートによって作成された HTTP プロファイルを共有できるようになっています。これによって、1 つのアプリケーションテンプレートによって作成された複数の環境で、似たような HTTP プロファイルを作成・保守しなくてもすみます。

ほとんどのアプリケーションテンプレートで作成される HTTP プロファイルは、共有が可能です。HTTP プロファイルが複数の環境で共有可能かどうかは、プロファイル名に **_shared_http** という文字列が付いているかどうかで見分けることができます。

たとえば、Microsoft IIS テンプレートを使って最初の環境を作成した際に、「**microsoft_iis_http_acceleration_shared_http**」という共有可能な HTTP プロファイルが作られたとします。その後は、同じような HTTP プロファイルを個別に作成するのではなく、Microsoft IIS テンプレートで作られたすべての環境でそのHTTPプロファイルを再利用することができます。

アプリケーションテンプレートオブジェクトの表示

テンプレートを使用し終えたら、そのテンプレートで作成されたローカルトラフィックオブジェクトが、そのオブジェクトタイプのリスト画面に表示されます。たとえば、アプリケーションテンプレートでロードバランシングプールを作成した場合、そのプールは設定ユーティリティの [Pools List] 画面に、BIG-IP での表示が許可されたほかのすべてのプールと共に表示されます。

同じテンプレートで作成されたオブジェクトは、ユーザ指定の同一のプレフィックスが名前の先頭に付いているため、アプリケーションに関連付けられたオブジェクトを見極めるのは簡単です。Microsoft IIS 環境を作成して、**IIS** というプレフィックスを指定した場合、その後は、同じテンプレートで作成されるすべてのオブジェクトの名前に

IIS プレフィックスが含まれることになります。たとえば、そのテンプレートで作成されるプールの名前は **IIS_pool**、バーチャルサーバの名前は **IIS_virtual_server** のようになります。

アプリケーションテンプレートオブジェクトの削除

ある特定のアプリケーションテンプレートを使用した後に、そのテンプレートで作成したオブジェクトをすべて削除するには、設定ユーティリティの対象のリスト画面にアクセスして、その画面に表示されたアプリケーションオブジェクトを削除する必要があります。上の例の続きで、Microsoft IIS テンプレートで作成したプールを削除するには、[Pools] リスト画面を表示して、**IIS_pool** オブジェクトを削除します。同様に、バーチャルサーバを削除するには、[Virtual Servers] リスト画面を表示して、**IIS_virtual_server** オブジェクトを削除します。すべての関連オブジェクトがシステムから削除されるまで、これを繰り返します。

◆ ヒント

設定ユーティリティでネットワークマップ機能を使えば、テンプレートに関連するオブジェクトをより簡単に特定することができます。

本ガイドについて

このガイドを使用する前に、BIG-IP 上でセットアップユーティリティを実行して、基本ネットワークと、スタティックおよびフローティングセルフ IP アドレス、インターフェイス、VLAN などのネットワーク要素の設定を行っておくことをお勧めします。詳細については、『***BIG-IP® Systems: Getting Started Guide***』を参照してください。

セットアップユーティリティを実行したら、設定ユーティリティを使用してバーチャルサーバ、ロードバランシングプール、プロファイルなどのローカルトラフィック管理オブジェクトを作成してさらにシステムをカスタマイズできます。

最後に、設定した要素を調整したり、必要に応じて新しい要素を追加します。

BIG-IP の設定をさらに調整したりカスタマイズしたりする前に、次の事項を終えておく必要があります。

- 設定ツールを選択する。
- 製品ガイドやオンラインヘルプなど、そのほかの参照資料をよく理解する。
- この章に記載されている表記規則を確認する。

補足情報の参照

このガイド以外にも、BIG-IP を使いこなすために役立つ資料がいくつかあります。以下の資料は Local Traffic Manager 製品に関連するものであり、F5 Prime Members Web サイトから PDF ファイルとしてダウンロードしていただけます。またこれらのガイドは、ブラウザベースの設定ユーティリティにアクセスしたときに、最初の Web ページからダウンロードすることも可能です。

- 『***BIG-IP® Systems: Getting Started Guide***』
  このガイドでは、BIG-IP の初期インストール、ライセンス取得、設定に必要なあらゆる情報を紹介しています。
- 『***TMOS™ Management Guide for BIG-IP® Systems***』
  このガイドには、BIG-IP のネットワークおよびシステム関連コンポーネントの設定および管理に必要なすべての情報が記載されています。このガイドを使用して、経路や VLAN の設定、自身の IP アドレスの割り当て、管理用ユーザアカウントの作成、冗長システムの管理などを行うことができます。
- 『***BIG-IP® Local Traffic Manager: Implementations***』
  このガイドでは、データ圧縮を使った SSL トラフィック処理や、リモート認証されるユーザアカウントへの特権の割り当てなど、具体的なゴールに向けた手順を紹介しています。『***Configuration Guide for BIG-IP® Local Traffic Management***』に含まれる詳細情報と関連付けられていることから、特定のトラフィック管理を設定するのに便利です。
- 『***Bigpipe Utility Reference Guide***』
  このガイドでは、**bigpipe** ユーティリティコマンドラインインターフェイスを使用するための構文情報を紹介しています。

- 『***Traffic Management Shell (tmsh) Reference Guide***』
  このガイドでは、**tmsh** ユーティリティコマンドラインインターフェイスを使用するための構文情報を紹介しています。

表記規則

重要な情報を容易に識別および理解できるように、本ドキュメントでは次のような表記規則を使用しています。

例の使用

本ドキュメント内のすべての例には、専用のクラス IP アドレスだけを使用しています。記載されている設定をセットアップする場合は、サンプルアドレスの代わりに、ユーザ自身のネットワークに適した有効な IP アドレスを使用してください。

新規用語の識別

定義されている用語を識別しやすいように、該当用語を太字の斜体で表記しています。例 : ***バーチャルサーバ***は、BIG-IP や他のホストサーバが管理しているコンテンツサイトに関連付けられた仮想アドレスと仮想ポートの特定の組み合わせを指します。

オブジェクト、名前、コマンドへの参照の識別

文中で識別しやすいように、さまざまな項目を太字で表記しています。Web アドレス、IP アドレス、ユーティリティ名、変数やキーワードといったコマンドの一部などが太字になっていることがあります。例 : **Idle Timeout** 値に **5** を設定できます。

他のドキュメントへの参照の識別

別のドキュメントまたは特定の項への参照は、斜体で表記しています。書籍タイトルへの参照は、太字の斜体で表記しています。例 : インストールについては、『***BIG-IP® Systems: Gettng Started Guide***』を参照してください。

コマンド構文の識別

完全なコマンドは、太字の Courier 文字で表示しています。コマンド行画面全体を表示するコマンドでない場合は、該当する画面プロンプトにこの表記規則は当てはまらないので注意してください。たとえば次のコマンドは、指定されたプール名の設定を示します。

```
bigpipe self <ip_address> show
```

もしくは

```
b self <ip_Address> show
```

表 1.4 では、コマンド行構文で使用されるその他の特別な表記規則について説明しています。

テキスト内の項目	説明
\	コマンドが次の行に続き、改行を入れずにコマンド全体を入力することを示します。
< >	ユーザ定義のパラメータを示します。たとえば、コマンドに **<your name>** とある場合、ユーザの名前を入力します。括弧は取ります。
|	コマンドを分割します。
[]	括弧内の構文がオプションであることを示します。
...	一連の項目を入力できることを示します。

***表 1.4** コマンド行構文の表記規則*

ヘルプおよびテクニカルサポートリソースへのアクセス

追加のテクニカルドキュメントおよび製品情報は、次の場所で入手できます。

- **ローカルトラフィック管理のオンラインヘルプ**
  設定ユーティリティには、画面ごとにオンラインヘルプがあります。オンラインヘルプから、画面上の各コントロールおよび設定の説明を表示できます。左側のナビゲーションペインの [Help] タブをクリックすると、オンラインヘルプが表示されます。
- **設定ユーティリティの [Welcome] 画面**
  設定ユーティリティの [Welcome] 画面には、次のような便利な Web サイトおよびリソースへのリンクが数多く含まれています。
  - F5 Networks Technical Support Web サイト（英語のみ）
  - F5 Solution Center（英語のみ）
  - F5 DevCentral Web サイト
  - プラグイン、SNMP MIB、および SSH クライアント
- **F5 Prime Members Web サイト**
  日本のお客様には、F5 Prime Members Web サイト（**https://www.f5networks.co.jp/primemembers/**）より、以下の最新製品ドキュメントを提供しています。
  - 各製品の新旧リリースノート
  - 各製品ガイド
  - The Ask F5[SM] Knowledge Base

  このサイトにアクセスするには、**http://www.f5networks.co.jp/f5pm** での登録が必要です。

2

バーチャルサーバの設定

- バーチャルサーバおよび仮想アドレスの概要
- バーチャルサーバのタイプについて
- クラスタリングマルチプロセッシングの概要
- バーチャルサーバの作成
- バーチャルサーバと仮想アドレスの設定について
- バーチャルサーバおよび仮想アドレスの管理

バーチャルサーバおよび仮想アドレスの概要

バーチャルサーバは、BIG-IP® ローカルトラフィック管理の構成において最も重要なコンポーネントです。バーチャルサーバを構成する場合、バーチャルサーババーチャルサーバと仮想アドレスという 2 つの BIG-IP オブジェクトを作成します。

バーチャルサーバとは

バーチャルサーバは BIG-IP 上のトラフィック管理オブジェクトであり、1 つの IP アドレスとサービスによって表わされます。外部ネットワークのクライアントは、バーチャルサーバにアプリケーショントラフィックを送信し、続いて、設定に従いそのトラフィックをダイレクトします。多くの場合、バーチャルサーバの主な目的は、内部ネットワークのサーバプールにトラフィック負荷を分散することです。バーチャルサーバを使用すると、クライアントリクエストを処理するためのリソースのアベイラビリティが向上します。

バーチャルサーバは、複数のサーバ間でトラフィックを分散させるだけでなく、トラフィック管理のニーズに応じて、さまざまなタイプのトラフィックを異なる方法で処理します。たとえば、BIG-IP を通過する HTTP リクエスト データの圧縮を有効にしたり、SSL 接続を復号化および再暗号化したり、SSL 証明書を検証することが可能です。バーチャルサーバは、TCP、UDP、HTTP、SSL、SIP、FTP などのトラフィックタイプごとに設定グループ全体を適用して、BIG-IP による該当トラフィックタイプの管理方法に影響を与えることができます。

バーチャルサーバでは、特定のトラフィックタイプのセッションパーシステンスを有効にすることも可能です。バーチャルサーバを通じて、HTTP、SSL、SIP、および MSRDP セッションなどのセッションのパーシステンスを設定できます。

さらに、バーチャルサーバは iRule（個々の接続を特定の方法で検査およびダイレクトすることを目的としたユーザ作成スクリプト）を適用できます。たとえば、TCP 接続のコンテンツで特定の文字列を検索し、その文字列が検出された場合には、接続を特定のプールまたはプールメンバに送信するようバーチャルサーバに指示する iRule を作成できます。

要約すると、バーチャルサーバでは次のことを行えます。

- 複数のサーバ間でクライアントリクエストを分散させ、サーバの負荷のバランスを取ります。
- 特定タイプのトラフィックに対して、各種動作設定を適用します。
- 特定タイプのトラフィックのパーシステンスを有効にします。
- ユーザ作成の iRules™ に従ってトラフィックをダイレクトします。

バーチャルサーバは、次のようなさまざまな用途に使用できます。

- ***トラフィックのロードバランシングプールへのダイレクト***
  標準バーチャルサーバ（***ロードバランシング***バーチャルサーバとも呼ばれます）は、クライアントトラフィックをロードバランシングプールにダイレクトするもので、最も基本的なタイプのバーチャルサーバです。バーチャルサーバを初めて作成したときは、

既存のデフォルトプールを割り当てます。それ以降、バーチャルサーバは、そのデフォルトプールにトラフィックを自動的にダイレクトします。

- **VLAN ノードと IP アドレスを共有する**
  関連付けられた VLAN 内のノードと同じ IP アドレスを共有するように、*フォワーディング（レイヤ 2）* バーチャルサーバを設定できます。この場合、追加の設定作業を行う必要があります。具体的には、ノードが存在する VLAN を含む VLAN グループを作成し、その VLAN グループにセルフ IP アドレスを割り当て、関連 VLAN でバーチャルサーバを無効にします。詳細については、『***TMOS™ Management Guide for BIG-IP® Systems***』の VLAN と VLAN グループの章を参照してください。
- **特定の宛先 IP アドレスにトラフィックを転送する**
  フォワーディング（IP） バーチャルサーバは、ロードバランスするプールメンバがないという点を除いて、その他のバーチャルサーバと同様です。バーチャルサーバは、クライアントリクエストで指定されている宛先 IP アドレスにパケットを直接転送するだけです。フォワーディングバーチャルサーバを使用して要求を最初に指定された宛先 IP アドレスにダイレクトする場合、これらの接続は、ほかのバーチャルサーバの場合と同様に BIG-IP によって追加、追跡、リープされます。フォワーディングバーチャルサーバの統計情報を表示することもできます。
- **HTTP トラフィックの処理速度を向上させる**
  パフォーマンス（HTTP） バーチャルサーバは、Fast HTTP プロファイルを関連付けるバーチャルサーバです。バーチャルサーバとプロファイルを併用すると、バーチャルサーバによる HTTP リクエストの処理速度が向上します。
- **レイヤ 4 トラフィックの処理速度を向上させる**
  パフォーマンス（レイヤ 4） バーチャルサーバは、Fast L4 プロファイルを関連付けるバーチャルサーバです。バーチャルサーバとプロファイルを併用すると、バーチャルサーバによるレイヤ 4 要求の処理速度が向上します。

バーチャルサーバを作成する際は、そのバーチャルサーバからの着信トラフィックの宛先として機能させる1つまたは複数のプールを指定します。また、汎用プロパティ、設定オプション、およびその他の割り当てリソース（iRules、セッションパーシステンスのタイプなど）も設定できます。

「*バーチャルサーバのタイプについて*」（2-5 ページ）では、作成可能なバーチャルサーバのタイプとその汎用プロパティ、設定オプション、およびリソースについて説明します。

仮想アドレスとは

仮想アドレスとは、バーチャルサーバに関連付ける IP アドレスです。たとえば、バーチャルサーバの IP アドレスとサービスが **10.10.10.2:80** である場合、IP アドレス **10.10.10.2** が仮想アドレスとなります。

バーチャルサーバと仮想アドレスの間には、1 対多数の関係を作成できます。たとえば、3 つのバーチャルサーバ **10.10.10.2:80**、**10.10.10.2:443**、および **10.10.10.2:161** を、同じ仮想アドレス **10.10.10.2** に作成できます。

仮想アドレスは、有効にしたり 、無効にしたり できます。仮想アドレスを無効にすると 、このアドレスに関連付けられたバーチャルサーバはいずれも 、着信ネット ワークト ラフィックを受け入れることができません。

バーチャルサーバを作成する場合、仮想アドレスは間接的に作成します。このような場合、BIG-IP では、仮想アドレスを MAC アドレスに関連付けます。これにより、BIG-IP は、この仮想アドレスに対する ARP（Address Resolution Protocol: アドレス解決プロトコル）要求に応答し、仮想アドレスに基づいて Gratuitous ARP 要求と応答を送信します。また、ごくまれに ARP アクティビティがシステムパフォーマンスに影響を及ぼすことがありますが、そのような場合、仮想アドレスに対する ARP アクティビティを無効にすることができます。これは、BIG-IP 上に非常に多数の仮想アドレスが定義されている場合にのみ発生する可能性があります。

バーチャルサーバの仕組みについて

外部ネットワーク上にあるクライアントのユーザが Web ブラウザ経由で HTTP トラフィック送信するという状況を検証してみると、バーチャルサーバの仕組みがよく分かります。この場合、BIG-IP は以下のプロセスで動作します。

1. ユーザが Web ブラウザに URL を入力して、接続を開始します。ブラウザがその URL を、BIG-IP で事前に作成しておいたバーチャルサーバのアドレスに分解します。このバーチャルサーババーチャルサーバのアドレスが要求の宛先 IP アドレスとなります。
2. BIG-IP が対応するバーチャルサーバ設定をチェックして、着信トラフィックの送信先となる Web サーバプールを決定します。
3. BIG-IP がプール設定をチェックして、内部サーバノードの選択に使用するロード バランシングアルゴリ ズムを決定します。
4. BIG-IP がこのアルゴリズムを使い、特定のサーバノードを選択します。
5. BIG-IP が、要求パケットの **Destination IP Address** ヘッダに含まれる宛先 IP アドレス（つまり、バーチャルサーバのアドレス）を、選択されたノードの実際のアドレスに変更します。この場合、パケットに含まれる送信元アドレス（つまり、その接続を開始したクライアントのアドレス）は変更されずに残ります。
6. BIG-IP が、選択されたサーバノードに着信パケットを送信します。
7. BIG-IP をデフォルトゲートウェイとして、サーバノードからクライアントに応答が送信される場合、応答は BIG-IP 経由で返され、逆変換が発生します。BIG-IP が、応答パケットの **Source IP Address** ヘッダに含まれる応答の実際の送信元 IP

アドレス（サーバノードのアドレス）を、バーチャルサーバのアドレスに変換します。これによって、応答の送信元アドレスと要求の宛先アドレスが一致することになります。これは、クライアントが応答を受け入れるのに必要な要件です。

要するに、通常 BIG-IP は*要求の宛先アドレス*をサーバノードの実際のアドレスに変換し、実際の*応答の送信元アドレス*をバーチャルサーバのアドレスに変換します。つまり、外部ネットワークのクライアントからは、内部サーバノードのプライベートクラス IP アドレスが決して見えないようになっています。

◆注

*常に BIG-IP 経由でサーバ応答が返されるようにするには、サーバのデフォルト経路を内部 VLAN のセルフ IP アドレスに設定する必要があります。サーバが BIG-IP とは別のネットワーク上に置かれており、この設定ができない場合は、SNAT を作成することが可能です。詳細については、第 14 章「**SNAT の設定**」を参照してください。*

バーチャルサーバのタイプについて

作成可能なバーチャルサーバには、2 つのタイプ、ホストバーチャルサーバとネットワークバーチャルサーバがあります。

ホストバーチャルサーバ

*ホストバーチャルサーバ*は、インターネット Web サイトや FTP サイトなどの特定のサイトを表し、プールのメンバであるコンテンツサーバをターゲットにしたトラフィックをロードバランスします。

ホストバーチャルサーバに割り当てる IP アドレスは、DNS（Domain Name System: ドメイン名システム）がサイトのドメイン名に関連付ける IP アドレスと一致する必要があります。BIG-IP は、そのサイトに対する接続要求を受信した場合、クライアントの宛先 IP アドレスがバーチャルサーバの IP アドレスと一致することを認識し、以降はバーチャルサーバがロードバランスするコンテンツサーバの1つにクライアントリクエストを転送するようになります。

ネットワークバーチャルサーバ

*ネットワークバーチャルサーバ*は、IP アドレスのホスト部分にビットが設定されていない（つまり、IP アドレスのホスト部分が **0** である）バーチャルサーバです。ネットワークバーチャルサーバには、2 種類あります。クライアントトラフィックを宛先 IP アドレスの範囲に基づいてダイレクトするものと、クライアントトラフィックを BIG-IP が認識しない特定の宛先 IP アドレスに基づいてダイレクトするものです。

特定範囲の宛先 IP アドレスに対するトラフィックのダイレクト

ホストビットが **0** に設定されている IP アドレスを使用すると、バーチャルサーバは、単一の宛先 IP アドレスではなく、全範囲の IP アドレスが宛先になっているクライアント接続をダイレクトできます（ホストバーチャルサーバの場合と同様です）。したがって、バーチャルサーバの IP アドレスによって指定されているネットワーク内の宛先 IP アドレスがクライアント接続のターゲットになっている場合、BIG-IP はその接続を、ネットワークバーチャルサーバに関連付けられている 1 つ以上のプールにダイレクトできます。

たとえば、バーチャルサーバは、**192.168.1.0** ネットワーク上のノードのいずれかが宛先になっているクライアントトラフィックを、特定のロードバランシングプール（**ingress-firewalls** など）にダイレクトできます。また、バーチャルサーバは、サブネット **192.168.1.0/24** 内の任意のアドレスが宛先になっている Web 接続を、プール **default_webservers** にダイレクトすることもできます。

透過デバイスのトラフィックのダイレクト（ワイルドカードバーチャルサーバ）

ネットワークバーチャルサーバには、特定のネットワークまたはサブネットが宛先になっているクライアント接続をダイレクトするだけでなく、バーチャルサーバによって認識されない特定の宛先 IP アドレス（透過デバイスなど）を持つクライアント接続もダイレクトできるものがあります。このタイプのネットワークバーチャルサーバは、*ワイルドカードバーチャルサーバ*と呼ばれます。

*ワイルドカードバーチャルサーバ*は、透過ネットワークデバイスをターゲットにしたネットワークトラフィックを管理するために設計された、特殊なタイプのネットワークバーチャルサーバです。透過デバイスの例としては、ファイアウォール、ルータ、プロキシサーバ、キャッシュサーバがあります。ワイルドカードバーチャルサーバは、BIG-IP で認識されていない宛先 IP アドレスがあるネットワークトラフィックを管理します。

認識されていないクライアント IP アドレスの処理

ホストタイプのバーチャルサーバは通常、特定サイトのトラフィックを管理します。対応するサイトの接続要求を受信すると、BIG-IP は、そのバーチャルサーバによってロードバランスされるコンテンツサーバの 1 つにクライアントを転送します。

ただし、透過ノードをロードバランスする際、BIG-IP はクライアントの宛先 IP アドレスを認識しないことがあります。クライアントは、ファイアウォール、ルータ、またはプロキシサーバの反対側の IP アドレスに接続する可能性があります。このような状況では、BIG-IP は、クライアントの宛先 IP アドレスをバーチャルサーバの IP アドレスと一致させることができません。

ワイルドカードネットワークバーチャルサーバは、BIG-IP 上のバーチャルサーバレベルでは着信 IP アドレスを変換しないようにすることで、この問題を解決します。たとえば、BIG-IP は、クライアントの宛先 IP アドレスと一致する特定のバーチャルサーバを検出しなかった場合、クライアントの宛先 IP アドレスを、IP アドレス **0.0.0.0** で指定されたワイルドカードバーチャルサーバと一致させます。次に、BIG-IP は、クライアントのパケットを、ワイルドカードバーチャルサーバがロードバランスするファイアウォールまたはルータの1つに転送します。その後、クライアントのパケットは、実際の宛先 IP アドレスに転送されます。

デフォルトのワイルドカードサーバとポート固有のワイルドカードサーバについて

作成可能なワイルドカードバーチャルサーバには、次の 2 種類があります。

- *デフォルトのワイルドカードバーチャルサーバ*
  *デフォルトのワイルドカードバーチャルサーバ*は、ポート **0** を使用し、すべてのサービスのトラフィックを処理するワイルドカードバーチャルサーバです。ワイルドカードバーチャルサーバは、デフォルトではすべての VLAN に対して有効になっています。ただし、デフォルトのワイルドカードバーチャルサーバでサポートする必要のない VLAN を無効にすることができます。デフォルトのワイルドカードバーチャルサーバに対して VLAN を無効にするに

は、VLAN 無効化リストを作成します。VLAN 無効化リストは、デフォルトのワイルドカードバーチャルサーバのみに適用されます。特定の VLAN のみに関連付けられているワイルドカードバーチャルサーバに対して VLAN 無効化リストを作成することはできません。デフォルトのワイルドカードサーバを作成する手順については、「*ワイルドカードバーチャルサーバの作成*」（2-12 ページ）を参照してください。

- **ポート固有のワイルドカードバーチャルサーバ**
  ポート固有のワイルドカードバーチャルサーバは、特定のサービスのみのトラフィックを処理します。このワイルドカードバーチャルサーバを定義する際は、サービス名またはポート 番号を使用します。ポート 固有のワイルドカードバーチャルサーバを使用すると、特定タイプのネットワークトラフィックの統計情報を追跡したり、発信トラフィック（ HTTPトラフィックなど）をファイアウォールやルータではなく、キャッシュサーバに直接ルーティングすることができます。ポート 固有のワイルドカードバーチャルサーバを作成する手順については、「 *ポート固有のワイルドカードバーチャルサーバを作成するには*」（ 2-13 ページ）を参照してください。

デフォルトのワイルドカードバーチャルサーバとポート固有のワイルドカードバーチャルサーバの両方を使用する場合、標準のバーチャルサーバともポート固有のワイルドカードバーチャルサーバのいずれとも一致しないトラフィックはすべて、デフォルトのワイルドカードバーチャルサーバによって処理されます。

複数タイプのサービスを処理する必要がある透過ノード（ファイアウォール、ルータなど）を定義する場合は、ノードの実際のポートを指定し、バーチャルサーバに対してポート変換をオフにすることをお勧めします。

複数のワイルドカードサーバの作成

同時に動作する複数のワイルドカードバーチャルサーバを定義できます。各ワイルドカードバーチャルサーバを個々の VLAN に割り当てる必要があるので、それぞれのワイルドカードバーチャルサーバは該当 VLAN のパケットのみを処理できます。

構成によっては、BIG-IP の片側にワイルドカードバーチャルサーバを設定して、透過デバイス間の接続をロードバランスする必要があります。BIG-IP の他方の側に別のワイルドカードバーチャルサーバを作成し、透過デバイスから接続を受信してその宛先に転送するバーチャルサーバにパケットを転送することができます。

クラスタリングマルチプロセッシングの概要

BIG-IP には、「クラスタリングマルチプロセッシング（CMP）」というパフォーマンス機能が実装されています。***CMP*** とは、システムの CPU ごとに Traffic Management Microkerne（TMM）サービスのインスタンスを個別に作成するトラフィック高速化機能です。CMP が有効になっている場合、作業負荷がすべての CPU で均等に共有されます。

バーチャルサーバを作成すると、BIG-IP で CMP 機能が自動で有効化されます。CMP が有効になっている場合、TMM サービスのすべてのインスタンスによってアプリケーショントラフィックが処理されます（バーチャルサーバの設定方法については、「*バーチャルサーバの作成*」（2-11 ページ）を参照してください）。

CMPが有効になっている場合のシステム動作の違いについて

CMP 機能は、あくまでもアプリケーショントラフィックを高速化するための内部パフォーマンス機能ですが、システムに対する効果が認められるケースもあります。図 2.1 は **bigpipe virtual show all** コマンドの出力であり、バーチャルサーバ **my_virtual** に対して CMP が有効になっていることを示しています。

```
[admin@silversalmon:Active] admin # b virtual show all | more
VIRTUAL ADDRESS 12.12.12.12   UNIT 1
|     ARP enable
|     (cur, max, limit, tot) = (0, 0, 0, 0)
|     (pkts,bits) in = (0, 0), out = (0, 0)
|     PVA (cur, max, limit, tot) = (0, 0, 0, 0)
|     PVA (pkts,bits) in = (0, 0), out = (0, 0)
|     PVA assist conns (tot, curr) = (0, 0)
+-> VIRTUAL my_virtual   SERVICE http
    |     PVA acceleration none
    |     CMP enable on none   mode: all
    |     (cur, max, limit, tot) = (0, 0, 0, 0)
    |     (pkts,bits) in = (0, 0), out = (0, 0)
    |     requests (total) = 0
    |     PVA (cur, max, limit, tot) = (0, 0, 0, 0)
    |     PVA (pkts,bits) in = (0, 0), out = (0, 0)
    |     PVA assist conns (tot, curr) = (0, 0)
    |     ephem (cur, max, limit, tot) = (0, 0, 0, 0)
    |     ephem (pkts,bits) in = (0, 0), out = (0, 0)
    |     no nodes errors = 0
   +-> POOL my_pool  LB METHOD round robin   MIN/CUR ACTIVE MEMBERS 0/0
       |     (cur, max, limit, tot) = (0, 0, 0, 0)
       |     (pkts,bits) in = (0, 0), out = (0, 0)
```

図 2.1 バーチャルサーバに対して CMP が有効になっていることを示す bigpipe 出力

設定ユーティリティを使って標準パフォーマンスグラフを表示すると、TMM サービスの複数のインスタンス（**tmm0** や **tmm1** など）を確認することができます。図 2.2（2-9 ページ）は、複数の CPU と複数の TMM インスタンスのパフォーマンスを示すパフォーマンスグラフです。

図 2.2 CMP が有効になったシステムのパフォーマンスグラフ

CMP が有効になっている場合は、次のことに注意してください。

- TMM インスタンスごとに複数の統計情報が個別に表示される一方で、すべての TMM インスタンスの合計として、別の統計情報が表示されます。
- CMP が有効になっている場合、バーチャルサーバのコネクションリミットが TMM サービスのすべてのインスタンスに均等に分散されます。
- ロードバランシングモードによっては、CMP が無効のシステムとは違う動作をするものがあります。

◆注

*プールメンバに対するコネクションリミットの設定値が小さな場合（8400 プラットフォームでコネクションリミットが **2**、または 8800 プラットフォームで **4** といった場合など）は、CMP 機能を無効にすることを推奨します。*

CMP の無効化と再有効化

場合によっては、TMM サービスを特定の CPU に割り当てたほうがいいケースがあります。これを行うには、ハードウェアプラットフォームが限定されているものの、CMP をまとめて無効にします。CMP を無効にするには、bigdb 変数 **provision.tmmcount** を設定して、**bigstart restart** コマンドを実行します。

```
bigpipe db provision.tmmcount 1
bigstart restart
```

CMP を再度有効にするには、以下に示すように、bigdb 変数の値を **1** から **0** に変更して、再び **bigstart restart** コマンドを実行します。

```
bigpipe db provision.tmmcount 0
bigstart restart
```

冗長システムの CMP について

冗長システムユニットで CMP 機能が有効になっている場合は、以下のようなシステム動作に注意するようにしてください。

- ハイアベイラビリティテーブル（HA）で、TMM サービスの各インスタンスが個別のエントリを持ちます。
- TMM サービスのインスタンスが 1 つでも停止すれば、アクティブユニットがフェイルオーバします。
- TMM サービスのすべてのインスタンスが同意した場合にのみ、VLAN フェイルセーフによるフェイルオーバがトリガされます。
- ハードウェアプラットフォームによっては、接続ミラーリングがサポートされる場合があります。

バーチャルサーバの作成

設定ユーティリティを使用して、バーチャルサーバを作成するか、既存のバーチャルサーバの設定を変更することができます。次の項では、バーチャルサーバを作成および変更する手順を説明します。個々のバーチャルサーバのプロパティおよび設定については、「*バーチャルサーバと仮想アドレスの設定について*」（2-14 ページ）を参照してください。既存のバーチャルサーバの設定の表示については、「*バーチャルサーバおよび仮想アドレスの管理*」（2-21 ページ）を参照してください。

バーチャルサーバを作成する場合、その設定に多くのデフォルト値を使用するバーチャルサーバを作成できます。これにより、明示的に設定すべき設定の数が大幅に削減されるため、バーチャルサーバを作成する作業が素早く、しかも簡単になります。

バーチャルサーバを作成する場合、そのタイプをホスト バーチャルサーバまたはネットワーク バーチャルサーバに指定できます（ホストおよびネットワーク バーチャルサーバの詳細については、「*ホストバーチャルサーバ*」（2-5 ページ）と「*ネットワークバーチャルサーバ*」（2-5 ページ）を参照してください）。いずれの場合も、設定する必要があるのは、バーチャルサーバの一意の名前と、宛先アドレスおよびサービスポート だけです。バーチャルサーバのタイプがネットワーク である場合、宛先タイプと ネット マスクも 設定する 必要があります。

◆重要

*バーチャルサーバが作成されると、BIG-IP はそのバーチャルサーバを現在の管理パーティションに配置します。パーティションの詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

バーチャルサーバを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Servers] 画面が開きます。
2. 画面の右上の [**Create**] ボタンをクリックします。
   [New Virtual Server] 画面が開きます。

   ***注**: **[Create]** ボタンをクリックできない場合は、現在のユーザロールではバーチャルサーバを作成する権限が与えられていないことを示します。*
3. 必要な設定を行います。
   詳細については、表 2.1（2-14 ページ）を参照してください。
4. 他の設定値をそのまま使用するか変更します。
   各設定の詳細については、「*バーチャルサーバと仮想アドレスの設定について*」（2-14 ページ）を参照してください。
5. [**Finished**] をクリックします。

◆注

冗長システムの構成では、最初にユニット 1 のバーチャルサーバを作成していないかぎり、ユニット 2 のバーチャルサーバを作成することはできません。

◆注

*関連付けられている VLAN 内のノードと同じ IP アドレスをバーチャルサーバで使用する場合には、追加の設定作業が必要になります。具体的には、ノードが存在する VLAN を含む VLAN グループを作成し、その VLAN グループにセルフ IP アドレスを割り当て、関連 VLAN でバーチャルサーバを無効にします。詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

ワイルドカードバーチャルサーバの作成

ワイルドカードバーチャルサーバは、特殊なタイプのネットワークバーチャルサーバです。ワイルドカードバーチャルサーバを作成するには、次の 3 つの作業を行う必要があります。

- まず、透過デバイスのアドレスを含むプールを作成します。
- 次に、（デフォルトまたはポート固有の）ワイルドカードバーチャルサーバを作成します。
- 最後に、バーチャルサーバごとにポート変換が無効になっていることを確認します。ポート変換は、デフォルトでは無効になっています。

次の手順では、設定ユーティリティを使用してこれらの作業を行う方法を説明します。ワイルドカードバーチャルサーバのタイプの詳細については、「*透過デバイスのトラフィックのダイレクト（ワイルドカードバーチャルサーバ）*」（2-6 ページ）を参照してください。

透過デバイスのプールを作成するには

透過デバイスのプールを作成するには、[Pools] 画面を表示して [**Create**] ボタンをクリックします。詳細については、第 4 章「*ロードバランシングプールの設定*」を参照してください。

デフォルト のワイルド カード バーチャルサーバを 作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Servers] 画面が開きます。
2. 画面の右上の [**Create**] ボタンをクリックします。
   [New Virtual Server] 画面が開きます。

   ***注**: **[Create]** ボタンをクリックできない場合は、現在のユーザロールではバーチャルサーバを作成する権限が与えられていないことを示します。*
3. 必要な設定をすべて行います。
   [**Destination Address**] ボックスに IP アドレス **0.0.0.0** を入力してください。また、ネットワークタイプのバーチャルサーバを選択している場合は、[**Mask**]ボックスにネットマスク**0.0.0.0** を入力してください。
4. [**Finished**] をクリックします。

ポート 固有のワイルド カード バーチャルサーバを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Servers] 画面が開きます。
2. 画面右上の [**Create**] をクリックします。
   [New Virtual Server] 画面が開きます。

   ***注**: [Create] ボタンをクリックできない場合は、現在のユーザロールではバーチャルサーバを作成する権限が与えられていないことを示します。*
3. [**Address**] ボックスで、ワイルドカードの IP アドレス **0.0.0.0** を入力します。
4. [**Service Port**] 設定で、ポート番号を入力するか、リストからサービス名を選択します。ポート **0** は、すべてのタイプのサービスを処理するワイルドカードバーチャルサーバを定義します。ポート番号を指定すると、ポート固有のワイルドカードバーチャルサーバが作成されます。ワイルドカードバーチャルサーババーチャルサーバは、指定されたポートのトラフィックのみを処理します。
5. [Resources] セクションの [**Default Pool**] 設定で、バーチャルサーバに適用する透過デバイスのプールを選択します。
6. [**Finished**] をクリックします。

ワイルド カード バーチャルサーバに対し てポート 変換をオフにするには

ワイルドカードポートを使用してワイルドカードバーチャルサーバを定義した後、そのバーチャルサーバに対してポート変換が無効になっていることを確認する必要があります。

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Servers] 画面が開きます。
2. [Name] カラムで、ポート変換をオフにするバーチャルサーバをクリックします。
   [Virtual Servers] 画面が開きます。
3. [**Enable Translation**] セクションで、[**Port**] ボックスがオフになっていることを確認します。

バーチャルサーバと仮想アドレスの設定について

バーチャルサーバとそのバーチャルサーバアドレスには、設定可能なプロパティおよび設定が数多くあり、これらを設定することでバーチャルサーバによるトラフィックの管理方法に影響を与えることができます。また、バーチャルサーバに特定のリソース（ロードバランシングプール、パーシステンスプロファイルなど）を割り当てることもできます。これらのプロパティ、設定、およびリソースを組み合わせて、バーチャルサーバまたはそのアドレスの定義を表します。また、そのほとんどにデフォルト値があります。バーチャルサーバを作成する際は、デフォルト値をそのまま使用することも、必要に応じて調整することもできます。

次の項では、バーチャルサーバおよび仮想アドレスを定義するプロパティ、構成設定、およびリソースをすべて列挙し、説明します。

バーチャルサーバの作成方法については、「*バーチャルサーバの作成*」（2-11 ページ）を参照してください。

バーチャルサーバの設定

設定ユーティリティで、バーチャルサーバの設定は、汎用プロパティ、構成設定（基本および詳細）、およびリソース（基本および詳細）の 3 つのカテゴリに分類されています。次の項では、これら 3 つのカテゴリに含まれる設定について説明します。

汎用プロパティ

バーチャルサーバを作成する際には、いくつかの汎用プロパティを定義します。表 2.1 に、これらの汎用プロパティを列挙して説明します。

プロパティ	説明	デフォルト値
Name	バーチャルサーバに割り 当てる一意の名前。このプロパティは必須です。	デフォルト値なし
Destination Type	作成するバーチャルサーバのタイプとその IP アドレス。選択したタイプが **network** の場合、このプロパティには IP アドレスのマスクも含まれます。バーチャルサーバのタイプの詳細については、「*バーチャルサーバのタイプについて*」（2-5 ページ）を参照してください。このプロパティは必須です。	**Host**
Destination Address	バーチャルサーバの IP アドレス。	デフォルト値なし
Destination Mask	ネットワークバーチャルサーバのネットマスク。このプロパティは、ネットワークバーチャルサーバのみに適用されるもので、必須です。ネットマスクは、ホストビットが実際のゼロであるかワイルドカード表現であるかを明確にします。	デフォルト値なし
Service Port	トラフィックをダイレクトする対象のサービス名またはポート番号。このプロパティは必須です。	デフォルト値なし

表 2.1 *バーチャルサーバの汎用プロパティ*

プロパティ	説明	デフォルト値
State	バーチャルサーバの状態（**Enabled** または **Disabled**）。オプションとして、特定の VLAN に対してバーチャルサーバを有効または無効にできます。バーチャルサーバを無効にした場合、そのバーチャルサーバは新しい接続要求を受け付けなくなります。ただし、**down** 状態になる前に、現在の接続は処理を終了できます。 **注：**VLAN が指定されていない場合、**Enabled** または **Disabled** 設定は、すべての VLAN に適用されます。	**Enabled**

表 2.1 *バーチャルサーバの汎用プロパティ（続き）*

構成設定

バーチャルサーバを作成する際には、多数の設定を構成できます。表 2.2 に、これらのバーチャルサーバの構成設定を列挙して説明します。これらの設定のすべてにデフォルト値が設定されています。これらの設定は、必要がなければ変更する必要はありません。

設定	説明	デフォルト値
Type	バーチャルサーバ設定のタイプ。選択肢は、[**Standard**]、[**IP Forwarding (IP)**]、[**Forwarding (Layer 2)**]、[**Performance (HTTP)**]、[**Performance (Layer 4)**]、および [**Reject**] です。詳細については、「*バーチャルサーバおよび仮想アドレスの概要*」（2-1 ページ）を参照してください。[**Reject**] に設定した場合、BIG-IP はバーチャルサーバの IP アドレスが宛先になっているトラフィックをすべて拒否します。	**Standard**
Protocol	バーチャルサーバでトラフィックをダイレクトする対象とするネットワークプロトコル名。サンプルプロトコル名は、**TCP** および **UDP** です。 この機能の利点の 1 つは、複数の VPN 間で仮想プライベートネットワーク（VPN）クライアント接続をロードバランスできるため、シングルポイント障害を防止できるということです。この機能の標準的な用途は、非変換バーチャルサーバを使用して、IPSEC VPN サンドイッチにおける複数の VPN ゲートウェイをロードバランスすることです。 注目すべき重要な点は、このようなプロトコルのアドレス変換は、オプションでアクティブにすることができますが、一部のプロトコル（AH モードの IPSEC など）は、変更されないままの IP ヘッダに依存するということです。そうような場合は、非変換ネットワークバーチャルサーバを使用する必要があります。 この設定は、パフォーマンス（HTTP）タイプのバーチャルサーバを作成する際には無効になっています。	**TCP**
Protocol Profile (Client)	選択されたプロファイルをクライアント側のプロファイルとして指定する設定。TCP 接続と UDP 接続のみに適用されます。パフォーマンス（HTTP）タイプのバーチャルサーバを作成する場合、この値は **fasthttp** に設定され、変更することはできません。同様に、パフォーマンス（レイヤ 4）タイプのバーチャルサーバを作成する場合、この値は **fastl4** に設定され、変更することはできません。詳細については、第 8 章「*プロトコルプロファイルの管理*」を参照してください。	**TCP**
Protocol Profile (Server)	選択されたプロファイルをサーバ側のプロファイルとして指定する設定。TCP 接続と UDP 接続のみに適用されます。この設定は、パフォーマンス（HTTP）またはパフォーマンス（レイヤ 4）タイプのバーチャルサーバを作成する際には表示されません。詳細については、第 8 章「*プロトコルプロファイルの管理*」を参照してください。	**(Use Client Profile)**

表 2.2 *バーチャルサーバの設定*

設定	説明	デフォルト値
OneConnect Profile	接続のパーシステンスを管理するための既存の OneConnect™ プロファイルの名前。この設定は、パフォーマンス（HTTP）またはパフォーマンス（レイヤ 4）タイプのバーチャルサーバを作成する際には表示されません。 ***重要**: OneConnect プロファイルの **Maximum Size** の設定は、バーチャルサーバのアベイラビリティに影響を及ぼします。詳細については、第 11 章「その他のプロファイルの使用」を参照してください。*	**oneconnect**
NTLM Conn Pool	既存の NTLM プロファイルの名前 OneConnect プロファイルと組み合わせて使用する場合、NTLM プロファイルは NT Lan Manager（NTLM）トラフィックのサーバ側の接続をプールします。	**None**
HTTP Profile	HTTP トラフィックを管理するための既存の HTTP プロファイルの名前。この設定は、パフォーマンス（HTTP）またはパフォーマンス（レイヤ 4）タイプのバーチャルサーバを作成する際には表示されません。詳細については、第 6 章「*アプリケーション層トラフィックの管理*」を参照してください。	**None**
FTP Profile	FTP トラフィックを管理するための既存の FTP プロファイルの名前。この設定は、パフォーマンス（HTTP）またはパフォーマンス（レイヤ 4）タイプのバーチャルサーバを作成する際には表示されません。詳細については、第 6 章「*アプリケーション層トラフィックの管理*」を参照してください。	**None**
SSL Profile (Client)	クライアント側の SSL トラフィックを管理するための既存の SSL プロファイルの名前。この設定は、パフォーマンス（HTTP）またはパフォーマンス（レイヤ 4）タイプのバーチャルサーバを作成する際には表示されません。詳細については、第 9 章「*SSL トラフィックの管理*」を参照してください。	**None**
SSL Profile (Server)	サーバ側の SSL トラフィックを管理するための既存の SSL プロファイルの名前。この設定は、パフォーマンス（HTTP）またはパフォーマンス（レイヤ 4）タイプのバーチャルサーバを作成する際には表示されません。詳細については、第 9 章「*SSL トラフィックの管理*」を参照してください。	**None**
Authentication Profile	認証メカニズムを管理するための既存の認証プロファイルの名前。例として、リモート LDAP または RADIUS サーバがあります。この設定は、パフォーマンス（HTTP）またはパフォーマンス（レイヤ 4）タイプのバーチャルサーバを作成する際には表示されません。詳細については、第 10 章「*アプリケーショントラフィックの認証*」を参照してください。	**None**
Stream Profile	データストリーム内（TCP 接続など）で文字列を検索および置換するときに使用する、既存の Stream プロファイルの名前。この設定は、パフォーマンス（HTTP）またはパフォーマンス（レイヤ 4）タイプのバーチャルサーバを作成する際には表示されません。詳細については、第 11 章「*その他のプロファイルの使用*」を参照してください。	**None**
RTSP	既存の RTSP プロファイルの名前 ***Real Time Streaming Protocol***（RTSP）はストリーミングメディアプレゼンテーション用のプロトコルであり、クライアントシステムは RTSP を使用することで、リモートのストリーミングメディアサーバを制御し、サーバ上のファイルへの時間ベースのアクセスを可能にします。	**None**
SMTP	既存の SMTP プロファイルの名前 SMTP プロファイルは Simple Mail Transport Protocol（SMTP）に関連するプロファイルです。この設定は Protocol Security Module のライセンスがある場合にのみ表示されます。詳細については、『***Configuration Guide for BIG-IP® Protocol Security Module***』を参照してください。	**None**

***表 2.2** バーチャルサーバの設定（続き）*

設定	説明	デフォルト値
SIP Profile	SIP トラフィックを管理する既存の SIP プロファイルの名前。詳細については、「*SIP プロファイルの設定*」（6-40 ページ）を参照してください。なお、[**SIP Profile**] オプションは、**Standard** タイプのバーチャルサーバでのみ指定できます。	**None**
Statistics Profile	統計情報プロファイルの名前。詳細については、第 11 章「*その他のプロファイルの使用*」を参照してください。	**stats**
VLAN Traffic	バーチャルサーバを有効または無効にする対象の VLAN の名前。VLAN の詳細については、『***TMOS® Management Guide for BIG-IP® Systems***』を参照してください。	**ALL VLANS**
Rate Class	着信ネットワークトラフィックのスループットポリシーを実施するために使用される、既存のレートクラスの名前。詳細については、第 16 章「*レートシェーピングの設定*」を参照してください。	**None**
Traffic Class	バーチャルサーバに割り当てるトラフィッククラスのリスト。トラフィッククラスで定義された基準を満たすあらゆるトラフィックフローに対して、分類 ID によるタグ付けが行われます。詳しくは第 15 章「*トラフィッククラスの設定*」を参照してください。	デフォルト値なし
Connection Limit	バーチャルサーバに許可された同時接続の最大数。**0** に設定すると、コネクションリミットがオフになります。	**0**
Connection Mirroring	冗長ペアのアクティブユニットから予備ユニットへ、接続をミラーリングする設定。この設定は、高い信頼性を保証しますが、システムパフォーマンスに影響を及ぼします。 ***重要***：*再起動処理の後も、予備ユニットがミラー接続を維持するため、パフォーマンス(レイヤ4)バーチャルサーバでのみコネクションミラーリングを有効にすることをお勧めします。(詳細については、この表の* ***Type*** *の設定と、「**バーチャルサーバとは」(2-1 ページ)**を参照してください)。また、コネクションミラーリングの専用帯域を確保するため、ピアユニット間にダイレクトリンク（トランク）を設定することをお勧めします。これにより、パフォーマンスの問題やミラーリングされた情報の損失を防ぐことができます。*	無効（チェックされていない）
Address Translation	BIG-IP 上のアドレス変換を有効または無効にする設定。このオプションは、BIG-IP が同じ IP アドレスを持つデバイスをロードバランスする際に便利です。複数サーバのループバックデバイス上で重複した IP アドレスが設定されている、nPath ルーティング設定でよく使用されます。	有効（チェックあり）
Port Translation	BIG-IP でのポート変換を有効または無効にする設定。バーチャルサーバを使用して任意のサービスへの接続をロードバランスする必要がある場合、バーチャルサーバのポート変換をオフにすると便利です。	有効（チェックあり）

表 2.2 *バーチャルサーバの設定（続き）*

設定	説明	デフォルト値
Source Port	接続の送信元ポートを保持するかどうかを指定します。指定できる値は次のとおりです。 **Preserve**: 特定のSNATからの送信元ポートがすでに使用されている場合を除き、その送信元ポート用に設定した値が保持されます。そのポートが使用されている場合は、別のポートが使われます。 **Preserve Strict**: その送信元ポート用に設定した値が保持されます。ポートが使用されている場合、接続の処理は行われません。ポートがほかの接続によって使用されている場合、システムはその送信元ポートを使用するものの、その送信元ポートを共有する接続のトラフィックを宛先サーバが区別することはできません。F5 では、以下のいずれかにあてはまる場合は、この設定の使用を制限することを推奨しています。 • ポートが UDP トラフィック用に設定されている。 • システムが nPath ルーティング用の設定になっている、もしくは透過モードで動作している（つまり、ほかのレイヤ 3 フィールドやレイヤ 4 フィールドの変換が行われない）。 • 仮想 IP アドレスとノードアドレスが 1 対 1 で割り 当てられている、もしくはクラスタリングマルチプロセッシング（CMP）が無効になっている。 **Change**: 送信元ポートが変更されます。この設定は、内部ネットワークアドレスを難読化するのに有効です。	**Preserve**
SNAT Pool	既存の SNAT プールをバーチャルサーバに割り当てるか、もしくは Automap 機能を有効にします。この設定を使用する場合、BIG-IP はバーチャルサーバを経由するすべてのオリジナル送信元 IP アドレスを、その SNAT プールのアドレスに自動でマップします。指定できる値は、**None**、**Auto Map**、または既存の SNAT プールの名前となっています。詳細については、第 14 章「*SNAT の設定*」を参照してください。	**None**
Clone Pool (Client)	侵入検知に使用される機能であり 、設定した場合、（アドレス変換の前に）バーチャルサーバによってクライアント 側のトラフィックが、指定されたクローンプールのメンバに複製されます。クローンプールは、通常のプールと 同じトラフィックを受信します。したがって、クローンプールを使って検知システムにトラフィックを *複製*することができます。 **Clone Pool (Server)** を設定することも可能です。	**None**
Clone Pool (Server)	侵入検知に使用される機能であり、設定した場合、（アドレス変換の後に）バーチャルサーバによってサーバ側のトラフィックが、指定されたクローンプールのメンバに複製されます。クローンプールは、通常のプールと同じトラフィックを受信します。したがって、クローンプールを使って検知システムにトラフィックを *複製*することができます。 **Clone Pool (Client)** を設定することも可能です。	**None**
Last Hop Pool	ラストホッププールを使用して、ラストホップルータに応答トラフィックをダイレクトする設定。この設定は **auto_lasthop** の設定を上書きします。複数のルータを使用して BIG-IP に接続を送信している場合は、**auto_lasthop** グローバル変数が有効になっていると（デフォルト）、接続は受信元ルータと同じルータを介して自動的に送信されます。**auto-lasthop** から 1 つ以上のルータを除外する場合、またはグローバル **auto_lasthop** が何らかの理由で無効になっている場合（SSL ゲートウェイがある場合など）は、ラストホッププールを代用できます（**auto_lasthop** が有効になっている場合も、ラストホッププールが優先されます）。 ラストホッププールを設定する前に、まずルータ内部アドレスを含むプールを作成する必要があります。	**None**

表 2.2 *バーチャルサーバの設定（続き）*

リソース

バーチャルサーバに各種のトラフィックプロファイルを割り当てる以外に、プール、iRule、および 2 つのパーシステンスプロファイルを割り当てることができます。バーチャルサーバに割り当てるプール、iRule、およびパーシステンスプロファイルは、*リソース*と呼ばれます。

ロードバランシングタイプのバーチャルサーバを作成した場合、バーチャルサーバに割り当てる必要があるリソースの 1 つに、デフォルトのロードバランシングプールがあります。*デフォルトプール*は、別のプールを指定する iRule が存在しない場合に、BIG-IP がトラフィックを送信するプールです。iRule を使用して、トラフィックをプールへダイレクトする場合、この iRule をリソースとしてバーチャルサーバに割り当てる必要があります。

表 2.3 に、ロードバランシングバーチャルサーバに割り当て可能な特定のリソースを列挙して説明します。

Resource	説明	デフォルト値
iRules	接続のロードバランシング中にバーチャルサーバで使用する既存の iRules のリスト。選択した iRules のすべてについて、バーチャルサーバ上で対応するプロファイルを設定する必要があります。たとえば、HTTP コマンドを含む iRule を指定する場合は、バーチャルサーバ上でデフォルトまたはカスタムの HTTP プロファイルを設定する必要があります。同様に、認証 iRule を実装する場合は、デフォルトまたはカスタムの認証プロファイルを設定する必要があります。 実装する iRule が **[iRules]** リスト内に表示されない場合、その iRule は存在しないので、まず作成する必要があります。**[iRules]** 設定が [New Virtual Server] 画面に表示されない場合は、ライセンスを確認してください。iRules の詳細については、第 17 章「*iRule の記述*」を参照してください。	デフォルト値なし
HTTP Class Profiles	接続のロードバランシング実行時にバーチャルサーバで使用する既存の HTTP Class プロファイルのリスト。詳細については、第 8 章「*プロトコルプロファイルの管理*」を参照してください。	デフォルト値なし
Default Pool	バーチャルサーバでデフォルトプールとして使用するプール名。iRule によって別のプールにトラフィックを送信するようダイレクトされないかぎり、ロードバランシングバーチャルサーバは、このプールにトラフィックを自動的に送信します。詳細については、第 4 章「*ロードバランシングプールの設定*」を参照してください。	デフォルト値なし
Default Persistence Profile	BIG-IP で使用するパーシステンスのタイプ。この設定は、Standard、Performance（HTTP）、Performance（Layer 4）のバーチャルサーバでのみ指定することができます。詳細については、第 7 章「*セッションパーシステンスの有効化*」を参照してください。	**None**
Fallback Persistence Profile	指定されたデフォルトパーシステンスを使用できない場合に、BIG-IP で使用するパーシステンスのタイプ。この設定で使用できるパーシステンスプロファイルは、Source Address Affinity と Destination Address Affinity の 2 種類となっており、ほかに **None** と指定することも可能です。この設定は、Standard、Performance（HTTP）、Performance（Layer 4）のバーチャルサーバでのみ指定することができます。詳細については、第 7 章「*セッションパーシステンスの有効化*」を参照してください。	**None**

表 2.3 *ロードバランシングバーチャルサーバに割り当てられるリソース*

仮想アドレスの設定

設定ユーティリティでは、仮想アドレスのプロパティおよび設定が表示されます。表 2.4 に、仮想アドレスの汎用プロパティおよび構成設定を列挙して説明します。

プロパティ	説明	デフォルト値
Address	バーチャルサーバの IP アドレス。サービスは含みません。	デフォルト値なし
Unit ID	このアドレスを適用する冗長ペアユニットの ID。	**1**
Availability	サービスチェックに基づいた仮想アドレスのアベイラビリティ。	デフォルト値なし
State	仮想アドレスの状態（**enabled** または **disabled**）。	**Enabled**
Advertise Route	BIG-IP がこの仮想アドレスをアドバンスドルーティングモジュールにアドバタイズするためのバーチャルサーバの条件。この設定は、[**Route Advertisement**] 設定が有効（チェックあり）になっている場合にのみ適用されます。指定できる値は次のとおりです。 **When any virtual server is available（いずれかのバーチャルサーバが利用可能な場合）** **When all virtual server(s) are available（すべてのバーチャルサーバが利用可能な場合）** **Always（常時）**	**When any virtual server is available**（いずれかのバーチャルサーバが利用可能な場合）
Connection Limit	この仮想アドレスで BIG-IP に許可されている同時接続の数。	**0**
ARP	仮想アドレスの ARP 要求を有効または無効にする設定。無効になっている場合、BIG-IP は、この仮想アドレスで他のルータが送信した ARP 要求を無視します。	有効（チェックされている）
Route Advertisement	この仮想アドレスへのルートをカーネルルーティングテーブルに挿入し、アドバンスドルーティングモジュールが該当するルートをネットワーク上の他のルータに再配分できるようにする設定。	有効（チェックされている）

***表 2.4** 仮想アドレスの汎用プロパティと構成設定*

バーチャルサーバおよび仮想アドレスの管理

一般に、バーチャルサーバおよび仮想アドレスを管理する際は、既存のバーチャルサーバまたは仮想アドレスの設定を表示する必要があります。また、場合によっては、バーチャルサーバを削除する必要もあります。

作成したバーチャルサーバで作業する場合、次のことを行えます。

- バーチャルサーバの設定を表示または変更する。
- 仮想アドレスの設定を表示または変更する。
- バーチャルサーバと仮想アドレスのステータスを表示する。
- バーチャルサーバまたは仮想アドレスを有効または無効にする。
- バーチャルサーバまたは仮想アドレスを削除する。

◆注

*ユーザロールとパーティションアクセス割り当てに基づいて、管理が許可されているバーチャルサーバと仮想アドレスだけを操作できます。**Administrator**、**Resource Administrator**、または **Manager** ロールが割り当てられているユーザアカウントにかぎり、バーチャルサーバと仮想アドレスを管理することができます。*

バーチャルサーバの設定の表示または変更

場合によっては、バーチャルサーバの設定を調整するか、新しいバーチャルサーバを作成するかを決定する必要があります。バーチャルサーバの設定を表示する場合、以下の処理を実行できます。

- バーチャルサーバのリストを表示する。
- バーチャルサーバのプロパティと設定を表示または変更する。
- バーチャルサーバのリソースを表示する。
- バーチャルサーバの統計情報を表示する。

バーチャルサーバのリストの表示

ユーザが表示を許可されている既存のバーチャルサーバのリストを表示できます。バーチャルサーバのリストを表示する場合は、設定ユーティリティによって、各バーチャルサーバに関する次の情報が表示されます。

- 状態
- バーチャルサーバの名前
- バーチャルサーバが存在するパーティション
- 宛先（仮想アドレス）
- サービスポート
- バーチャルサーバのタイプ
- リソース（関連付けられたプール、HTTP Class プロファイル、iRules、およびパーシステンスプロファイル）

◆注

BIG-IP GTM が稼働し、かつリスナが設定されている場合、そのリスナはBIG-IP LTM のバーチャルサーバの一覧にバーチャルサーバとして表示されます。

バーチャルサーバのリストを表示するには

ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
これで、[Virtual Servers] 画面が開き、バーチャルサーバのリストが表示されます。

バーチャルサーバプロパティの表示または変更

バーチャルサーバのプロパティ（バーチャルサーバに割り当てられているプロファイルタイプなど）を表示できます。ただし、表示できるのは、ユーザが表示を許可された既存のバーチャルサーバのプロパティだけです。

バーチャルサーバプロパティを表示または変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Servers] 画面が開きます。
2. [Name] カラムで、バーチャルサーバの名前をクリックします。
   選択したバーチャルサーバのプロパティが表示されます。
3. バーチャルサーバのプロパティを変更するには、以下の手順を実行します。

   a) 画面でプロパティを見つけ、その値を変更します。

   b) [**Update**] をクリックします。

バーチャルサーバリソースの表示または変更

バーチャルサーバへのリソースとして割り当てられているデフォルトプール、デフォルトのパーシステンスプロファイル、およびフォールバックパーシステンスを表示できます。また、バーチャルサーバに関連付けられている iRule もすべて表示できます。これらのリソースを表示する手順は、次のとおりです。

バーチャルサーバリソースを表示または変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Servers] 画面が開きます。
2. バーチャルサーバの名前をクリックします。
   選択したバーチャルサーバのプロパティが表示されます。
3. メニューバーで [**Resources**] をクリックします。
   選択したバーチャルサーバの追加の設定が表示されます。
4. [Load Balancing] セクションで、バーチャルサーバのリソースをそのまま使用するか変更します。

5. [**Update**] をクリックします。
6. iRule またはHTTP Classプロファイルの割り当てを変更する場合は、対応する [**Manage**] ボタンをクリックします。
7. 既存の iRule または HTTP Class プロファイルを有効化 / 無効化するには、[**Move**] ボタン（**<<** または **>>**）を使用します。
8. [**Finished**] をクリックします。

バーチャルサーバの統計情報の表示

設定ユーティリティを使用して、既存のバーチャルサーバの統計情報を表示できます。

バーチャルサーバの統計情報を表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Server] 画面が開きます。
2. [Name] カラムで、バーチャルサーバの名前をクリックします。
3. [Statistics] メニューから [Virtual Server] を選択します。
   バーチャルサーバの統計情報が表示されます。

仮想アドレスの設定の表示または変更

仮想アドレス設定を表示または変更したい場合があります。このような場合、以下のように、仮想アドレスの設定を操作することができます。

- 仮想アドレスのリストを表示する。
- 仮想アドレスプロパティを表示または変更する。
- 仮想アドレス統計情報を表示する。

仮想アドレスのリストの表示

作成した既存の仮想アドレスのリストを表示して、任意の設定を調整できます。仮想アドレスのリストを表示する場合は、設定ユーティリティによって、そのアドレスの状態（**enabled** または **disabled**）も表示されます。

仮想アドレスのリストを表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Servers] 画面が開きます。
2. メニューバーで [**Virtual Address List**] をクリックします。
   既存の仮想アドレスのリストが表示されます。

仮想アドレスプロパティの表示または変更

仮想アドレスのプロパティを表示する手順は、次のとおりです。

仮想アドレスプロパティを表示または変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Servers] 画面が開きます。
2. メニューバーで [**Virtual Address List**] をクリックします。
   既存の仮想アドレスのリストが表示されます。
3. [Address] カラムで仮想アドレスをクリックします。
   選択した仮想アドレスのプロパティが表示されます。
4. 仮想アドレスのプロパティを変更するには、以下の手順を実行します。

   a) 画面でプロパティを見つけ、その値を変更します。

   b) [**Update**] をクリックします。

仮想アドレスの統計情報の表示

設定ユーティリティを使用して、既存の仮想アドレスの統計情報を表示できます。

仮想アドレスの統計情報を表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Servers] 画面が開きます。
2. メニューバーで [**Virtual Address List**] をクリックします。
   既存の仮想アドレスのリストが表示されます。
3. [Statistics] メニューから [**Virtual Address**] を選択します。
   仮想アドレスの統計情報が表示されます。

バーチャルサーバと仮想アドレスのステータスについて

設定ユーティリティを使用して、バーチャルサーバまたは仮想アドレスのステータスをいつでも確認できます。この情報は、バーチャルサーバまたは仮想アドレスのリストを表示して、[Status] カラムを確認するか、オブジェクトの **Availability** プロパティを確認することで把握できます。

設定ユーティリティでは、複数のアイコンのいずれかでステータスを示します。これらのアイコンは、色や形で区別できます。

- アイコンの*形*は、該当するノードでモニタからレポートされたステータスを示します。
- アイコンの*色*は、ノードの実際のステータスを示します。

ステータスについてこれらのアイコンを確認するには、表 2.5 を参照してください。設定ユーティリティ内でアイコンを表示するには、このページの「*バーチャルサーバプロパティを表示または変更するには*」（2-22 ページ）および「*仮想アドレスプロパティを表示または変更するには*」（2-24 ページ）参照してください。

ステータス インジケータ	説明
	バーチャルサーバまたは仮想アドレスが有効になっており、トラフィックを受け入れることができます。
	バーチャルサーバまたは仮想アドレスが有効になっていますが、現在利用できません。ただし、ユーザが処理を実行しなくても、バーチャルサーバまたは仮想アドレスが後で使用可能になる場合があります。 バーチャルサーバまたは仮想アドレスがこのステータスを示す例として、オブジェクトのコネクションリミットを超えている場合があげられます。接続数が設定した制限以下になると、バーチャルサーバまたは仮想アドレスが再び利用可能になります。
	バーチャルサーバまたは仮想アドレスが有効になっていますが、関連付けられたオブジェクトでこのバーチャルサーバまたは仮想アドレスが利用不可とマークされたため、オフラインになっています。バーチャルサーバまたは仮想アドレスがトラフィックを受信できるようにステータスを変更するには、バーチャルサーバまたは仮想アドレスをアクティブに有効化する必要があります。
	バーチャルサーバまたは仮想アドレスは稼働していますが、**Disabled** に設定されています。通常の動作に復帰するには、手動でバーチャルサーバまたは仮想アドレスを有効にする必要があります。
	バーチャルサーバまたは仮想アドレスのステータスが不明です。

***表 2.5** バーチャルサーバおよび仮想アドレスに対するステータスアイコンの説明*

バーチャルサーバまたは仮想アドレスの有効化 / 無効化

設定ユーティリティを使用して、バーチャルサーバまたは仮想アドレスをいつでも有効または無効にできます。バーチャルサーバまたは仮想アドレスを無効にすると、BIG-IP では、そのバーチャルサーバまたは仮想アドレスをターゲットとしたトラフィックを処理できなくなります。

まず最初に対応するリストを設定ユーティリティの画面に表示して、バーチャルサーバまたは仮想アドレスの有効 / 無効を切り替えることができます。

バーチャルサーバを有効または無効にするには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Server] 画面が開きます。
2. [Name] カラムで、有効または無効するバーチャルサーバの名前をクリックします。

3. [**State**] 設定を見つけます。
この設定は、バーチャルサーバが現在有効になっているか、無効になっているかを示します。
4. リスト画面に戻ります。
5. [Select] カラムで、バーチャルサーバ名に対応するボックスをクリックします。
6. [**Enable**] または [**Disable**] をクリックします。

仮想アドレスを有効または無効にするには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
[Virtual Server] 画面が開きます。
2. メニューバーで [**Virtual Address List**] をクリックします。
既存の仮想アドレスのリストが表示されます。
3. [Address] カラムで、有効または無効にする仮想アドレスを見つけます。
[State] カラムは、仮想アドレスが現在有効になっているか、無効になっているかを示します。
4. [Select] カラムで、仮想アドレスに対応するボックスをクリックします。
5. [**Enable**] または [**Disable**] をクリックします。

バーチャルサーバまたは仮想アドレスの削除

バーチャルサーバや仮想アドレスは、設定から永久的に削除できます。バーチャルサーバを削除すると、対応する仮想アドレスは、他のバーチャルサーバがその仮想アドレスに関連付けられていない場合、自動的に削除されます。

バーチャルサーバを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
[Virtual Server] 画面が開きます。
2. 削除するバーチャルサーバの左側にある [Select] ボックスをチェックします。
3. [**Delete**] をクリックします。
[Delete Confirmation] 画面が表示されます。
4. [**Delete**] をクリックします。
これでバーチャルサーバが削除されました。

仮想アドレスを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
[Virtual Server] 画面が開きます。

2. メニューバーで [**Virtual Address List**] をクリックします。
3. 削除する仮想アドレスの左側にある [Select] ボックスをチェックします。
4. [**Delete**] をクリックします。
   [Delete Confirmation] 画面が表示されます。
5. [**Delete**] をクリックします。
   これで仮想アドレスが削除されます。

3

ノードの設定

- ノードの概要
- ノードの作成と変更
- ノードの設定
- ノードの管理

ノードの概要

ノードは、BIG-IP の論理オブジェクトで、ネットワークの物理リソースの IP アドレスを特定します。ノードは明示的に作成することも、ロードバランシングプールにプールメンバを追加する際に BIG-IP で自動作成するように設定こと も可能です。

ノードとプールメンバとの違いは、ノードはデバイスの IP アドレスのみ（**10.10.10.10**）で指定されますが、プールメンバの宛先には IP アドレスとサービス（**10.10.10:80** など）が含まれるという点です。

ノードの主要な機能は、ヘルスモニタとの関連付けです。サーバのステータスを判定する方法として、プールメンバと同様に、ノードをヘルスモニタに関連付けることができます。プールメンバのヘルスモニタは、デバイス上で実行中のサービスのステータスをレポートしますが、ノードに関連付けられているヘルスモニタは、デバイス自体のステータスをレポートします。

たとえば、ICMP ヘルスモニタがプールメンバ **10.10.10.10:80** に対応するノード**10.10.10.10**に関連付けられており、モニタがノードを**down** 状態としてレポートする場合、モニタはプールメンバも **down** 状態としてレポートします。反対に、モニタがノードを **up** 状態としてレポートする場合は、実行中のサービスのステータに応じて、モニタはプールメンバを **up** または **down** のいずれかとしてレポートします。

設定ユーティリティを使用してノードを作成した後、必要に応じて設定を調整します。同じユーティリティを使用して、ノードに関する情報を表示したり、ノードを有効または無効にしたり、ノードを削除することが可能です。

ノードの作成と変更

ノードは、ロードバランシングプールを作成するための基礎となります。ロードバランシングプールに含める任意のサーバについて、まずノードを作成する（そのサーバをノードとして指定する）必要があります。サーバをノードとして指定した後、そのノードをプールメンバとしてプールに追加できます。また、ヘルスモニタをノードに関連付けて、そのサーバのステータスをレポートすることもできます。ロードバランシングプールへのノードの追加については、第 4 章「*ロードバランシングプールの設定*」を参照してください。

◆注

最初に対応するノードを作成せずにプールメンバを作成すると、BIG-IP が自動的にノードを作成します。

ノードを作成するには、設定ユーティリティを使用します。ノードを作成すると、BIG-IP によって、デフォルト設定のグループがそのノードに自動的に割り当てられます。これらのデフォルト設定は、そのまま使用することも、変更することもできます。また、ノードを作成した後でいつでも設定を変更できます。これらの設定については、「*ノードの設定*」(3-4 ページ)またはオンラインヘルプを参照してください。

BIG-IP の設定には、基本設定と詳細設定があります。ノードの作成時にデフォルト設定のいくつかを変更する場合は、必ず画面上で **[Advanced]** オプションを選択して、構成可能な設定をすべて表示するようにしてください。基本設定および詳細設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。

◆重要

*ノードが作成されると、BIG-IP はそのノードを現在の管理パーティションに配置します。パーティションの詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

ノードを作成するには

1. ナビゲーションペインの [Main] タブで、**[Local Traffic]** を展開して **[Nodes]** をクリックします。
   [Nodes] 画面が開きます。
2. 画面右上の **[Create]** をクリックします。
   [New Node] 画面が開きます。

   ***注**: [Create] ボタンをクリックできない場合は、現在のユーザロールではノードを作成する権限が与えられていないことを示します。*
3. **[Address]** 設定で、ノードの IP アドレスを入力します。
4. その他の設定のそれぞれを指定するか、そのまま使用するか、または変更します。
5. **[Finished]** をクリックします。

既存のノードを変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Nodes**] をクリックします。
   [Nodes] 画面が開きます。
2. [Address] カラムでアドレスをクリックします。
   選択したノードの設定が表示されます。
3. ノードの設定をそのまま使用あるいは変更します。
4. [**Update**] をクリックします。

ノードの設定

ノードの設定を行って、必要に応じてノードを調整できます。デフォルト値を持つ設定については、デフォルト設定をそのまま使用することも変更することもできます。また、ノードの設定は、ノードの作成時だけでなく、ノードの作成後にいつでも変更することができます。

表 3.1 に、ノードの構成可能な設定とそのデフォルト値を列挙します。この表の後に、特定の設定について説明します。

ノードの設定	説明	デフォルト値
Address	ノードの IP アドレスを指定します。この設定は必須です。	デフォルト値なし
Name	ノードの名前を指定します。	デフォルト値なし
Health Monitors	BIG-IPでデフォルトモニタとノードを関連付けるか、ユーザがノードに特定のモニタを割り当てるかを定義します。	**Node default**
Select Monitors	BIG-IP でノードに関連付けるモニタを指定します。この設定は、[**Health Monitors**] を [**Node Specific**] に設定した場合にのみ使用できます。	デフォルト値なし
Availability Requirement	BIG-IP によってノードが **up** 状態としてレポートされる前に、ノードをトラフィック受信可能としてレポートする必要があるヘルスモニタの最小数を指定します。この設定は、[**Health Monitors**] を [**Node Specific**] に設定した場合にのみ使用できます。	**すべて**
Ratio	ノードに割り当てる重み付けを指定します。	**1**
Connection Limit	ノードで許可された同時接続の最大数を指定します。	**0**

表 3.1 *ノードの構成設定*

ノードを設定する前に、変更する可能性がある特定のノード設定に関する説明を読んでおくと役立ちます。

ノードのアドレスの指定

設定するノードごとに、IP アドレスを指定する必要があります。ノード IP アドレスの例は、**10.10.10.10** です。この設定のみ必須です。

ノード名の指定

設定するノードごとに、ユニークなノード名をつけることができます（**Node_1** など）。ノード名は大文字小文字を区別し、文字、数字、および下線（_）のみを含めることができます。予約済みキーワードは使用できません。

ヘルスモニタの割り当て

BIG-IP を使用すると、モニタをこれらのノードに関連付けることで、ノードの健全性またはパフォーマンスを監視できます。これは、モニタをロードバランシングプールに関連付ける場合と同様です。ただし、ノードの場合は IP アドレスを監視しますが、プールの場合、プールメンバでアクティブになっているサービスを監視します。

BIG-IP には、監視対象のトラフィックタイプに応じて、ノードに関連付けることができる各種の設定済みモニタが含まれています。独自のカスタムモニタを作成してノードに関連付けることも可能です。ノードに関連付けることができない唯一の設定済みモニタは、ノードの監視用に設計されたモニタではなく、プールまたはプールメンバを監視するために特に設計されたモニタです。

◆注

*ノードと関連付けられるモニタは、かならず**Common** パーティションか、そのノードが含まれるパーティションのいずれかに配置されている必要があります。*

モニタをノードに関連付けるには、2 通りの方法があります。同じモニタ（すなわち、デフォルトのモニタ）を同時に複数のノードに関連付ける方法と、作成する各ノードにモニタを明示的に関連付ける方法です。

◆注

***bigpipe** ユーティリティコマンド **node** および **save** を使用して同じモニタを複数のノードに割り当てる場合（ たとえば、**bigpipe node all monitor icmp** および **bigpipe save**）、BIG-IP によって、ノードごとに別個のモニタとノードのエントリが **bigip.conf** ファイル内に作成されます。*

ヘルスモニタおよびパフォーマンスモニタの詳細については、第 12 章「*モニタの設定*」を参照してください。

デフォルトモニタの指定

この章で前述したように、最初に親ノードを作成せずにプールメンバを作成すると、BIG-IP が自動的に親ノードを作成します。作成する各ノードに 1 つ以上のモニタタイプを自動的に割り当てるように、BIG-IP を設定できます。これにより、作成する各ノードにモニタを明示的に選択する必要がなくなります。

デフォルトで、すべてのノードに 1 つ以上のモニタを関連付けるには、まず最初に、ノードに割り当てるモニタを指定する必要があります（次の「*1 つ以上のデフォルトモニタを割り当てるには*」を参照）。この処理を実行すると、自動的に作成されるすべてのノードに、指定されたデフォルトモニタが割り当てられます。

1 つ以上のデフォルト モニタを割り 当てるには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Nodes**] をクリックします。
   [Nodes] 画面が開きます。
2. メニューバーで [**Default Monitor**] をクリックします。
3. [**Health Monitors**] 設定で、[**Available**] ボックスからヘルスモニタを選択します。
4. [**Move**] ボタン (**<<**) をクリックして、モニタ名を [**Active**] ボックスに移動します。
5. デフォルトモニタに指定する各モニタに、同じ手順を繰り返します。
6. [**Availability Requirement**] リストから、次のいずれかを実行します。
   a) [**All**] を選択する。
      これにより、ノードが **up** であると見なされる前に、すべてのアクティブなモニタが正しく成功する必要があることが指定されます。
   b) [**At Least**] を選択し、数値を入力する。
      これにより、ノードが **up** であると見なされる前に、設定した数のモニタが正しく成功する必要があることが指定されます。
7. [**Update**] をクリックします。

以下のことを念頭に置いて、デフォルトモニタを使用するようにしてください。

- **Common** パーティション内のオブジェクトの管理権限を持つユーザが、ノードのデフォルトモニタに指定されたモニタ（**icmp** モニタなど）を無効にした場合、システムのすべてのノードが影響を受けます。ノードのデフォルトモニタは常に **Common** パーティションに配置するようにしてください。
- デフォルトモニタを指定するには、ユーザアカウントに **Administrator** ロールが割り当てられている必要があります。

- すべてのノードが同じパーティションに置かれている場合、デフォルトモニタはそのパーティション、もしくは **Common** パーティションに配置する必要があります。ノードが別々のパーティションに置かれている場合は、デフォルトモニタは **Common** パーティションに配置するようにします。

モニタのノードへの明示的関連付け

BIG-IP で自動的にノードを作成するのではなく、ユーザが明示的にノードを作成する場合、ノードを作成し、その **[Health Monitors]** を設定するときに、以下のいずれかを実行できます。

- **他のモニタをノードに関連付ける**
  他のモニタ（すなわち、デフォルト以外のモニタ）をノードに関連付けるには、ノードの設定を作成または変更するときに、ノードの **[Health Monitors]** 値を **Node Specific** に設定します。その後、設定ユーティリティを使用して、そのノードに関連付けることができるモニタをリストから選択できます。
- **デフォルトモニタをノードに関連付ける**
  デフォルトモニタをノードに関連付けるには、**[Health Monitors]** の値を **Node Default** に設定します。

アベイラビリティ要件の指定

[Availability Requirement] を設定すると、BIG-IP によってノードが **up** 状態としてレポートされる前に、ノードをトラフィック受信可能としてレポートする必要があるヘルスモニタの最小数を指定できます。指定可能な値は、**[All]** または指定の数値です。**[At Least]** 値を選択した場合は、数値を指定します。

重み付けの指定

[Ratio] 設定では、ノードの重み付けを指定します。デフォルトの設定は **1** です。重み付けについては、第 4 章「*ロードバランシングプールの設定*」を参照してください。

コネクションリミットの設定

[Connection Limit] 設定では、ノードに許可されている同時接続の最大数を指定できます。デフォルト値 **0**（ゼロ）を指定した場合、ノードで受信可能な同時接続の数に制限はありません。

ノードの管理

ノードを作成し、必要に応じて設定を行った後、追加の管理作業が必要となる場合があります。設定ユーティリティを使用して、次のことを行えます。

- ノードのリストを表示する
- ノードプロパティを表示する
- ノードステータスを表示し、確認する
- 既存のノードを有効または無効にする
- 既存のノードを削除する
- モニタの関連付けを無効にする

◆注

ユーザロールとパーティションアクセス割り当てに基づき、管理が許可されているノードのみを管理することができます。

ノードのリストを表示する

表示が許可されている既存のノードの一覧を表示することができます。ノードのリストを表示する場合、設定ユーティリティによって、各ノードに関する次の情報が表示されます。

- 状態
- ノードアドレス
- ノードが配置されたパーティション
- ノード名

BIG-IP で定義されたノードのリストを表示するには、次の手順に従います。

ノード のリスト を表示するには

ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Nodes**] をクリックします。
これで、[Nodes] 画面が開き、ノードのリストが表示されます。

ノードプロパティの表示

ノードの汎用プロパティを表示するには、設定ユーティリティを使用します。これらのプロパティとその説明を以下に示します。

- **アドレス**
  ノードの IP アドレス。
- **パーティション**
  ノードが配置されたパーティション
- **Availability**
  ノードのステータス。

- **Health monitors**
  ノードに関連付けられたヘルスモニタ。
- **Current connections**
  ノードが受信した現在の接続数。
- **State**
  ノードで受信するトラフィックの状態。指定できる値は以下のとおりです。
  - **Enabled（すべてのトラフィックを許可）**
  - **Disabled（パーシステンスまたはアクティブ接続だけを許可）**
  - **Forced offline（アクティブ接続だけを許可）**

ノード プロパティを表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Nodes**] をクリックします。
   [Nodes] 画面が開きます。
2. [Address] カラムで、表示するノードのアドレスをクリックします。
   選択したノードの設定が表示されます。

ノードステータスについて

設定ユーティリティを使用して、ノードのステータスをいつでも確認できます。この情報は、ノードのリストを表示して、[Status] カラムを確認するか、ノードの **Availability** プロパティを確認することで把握できます。

設定ユーティリティでは、複数のアイコンのいずれかでステータスを示します。これらのアイコンは、色や形で区別できます。

- アイコンの*形*は、該当するノードでモニタからレポートされたステータスを示します。
- アイコンの*色*は、ノードの実際のステータスを示します。

ノード ステータスについてこれらのアイコンを確認するには、表 3.2（ 3-10 ページ）を参照してください。設定ユーティリティ内でアイコンを表示するには、*ノードプロパティを表示するには*参照してください。

◆ヒント

*ノードのアベイラビリティを手動で設定するには、関連するヘルスモニタの **Manual Resume** 属性を設定します。詳細については、第 12 章「**モニタの設定**」を参照してください。*

.

ステータス インジケータ	説明
	ノードが有効で、トラフィックを受信できます。
	ノードは有効ですが、現在使用できません。ユーザが処理を実行しなくても、ノードが後で使用可能になる場合があります。使用できないノードが自動的に使用可能になる例としては、ノードへの同時接続数が、ノードに定義された **[Connection Limit]** 設定の値を下回った場合が挙げられます。
	ノードは有効ですが、関連付けられたモニタがノードを **down** としてマークしたため、オフラインになっています。ノードがトラフィックを受信できるように、ステータスを変更するには、ユーザの介入が必要です。
	ノードは **[Disabled]** に設定されていますが、モニタはノードを **up** としてマークしています。通常の動作に復帰するには、手動でノードを有効にする必要があります。
	ノードが **[Disabled]** に設定されており、**down** 状態です。通常の動作に復帰するには、手動でノードを有効にする必要があります。
	ノードは **[Disabled]** に設定されており、ユーザが無効にしたか、モニタがノードを **up** としてマークしたため、オフラインになっています。通常の動作に復帰するには、手動でノードを有効にする必要があります。
	ノードのステータスは不明です。不明のノードステータスには、以下のような原因があります。 ノードにモニタが関連付けられていない。 モニタの結果がまだ出力されていない。 ノードの IP アドレスの設定が誤っている。 ノードがネットワークから切断されている。

***表 3.2** ノードのステータスアイコンの説明*

ノードの有効化または無効化

トラフィックを受け付けるためには、ノードを有効にする必要があります。ノードが無効になっている場合、BIG-IP では既存の接続をタイムアウトにするか正常終了することができます。その場合、接続が既存のパーシステンスセッションに属している場合のみ、ノードは新しい接続を受け付けることが可能です。（無効化されたノードは、この点において、[**down**] に設定されているノードと異なります。**down** ノードでは、既存の接続をタイムアウトにすることはできますが、新しい接続はいっさい受け付けません）。

ノードを有効または無効にするには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Nodes**] をクリックします。
   [Nodes] 画面が開きます。
2. 有効または無効にするノードのアドレスを見つけます。
3. 左側のカラムで [Select] ボックスをチェックします。
4. 画面の下部で、[**Enable**] または [**Disable**] をクリックします。

ノードの削除

プール内で使用しなくなったノードは削除できます。

ノードを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Nodes**] をクリックします。
   [Nodes] 画面が開きます。
2. 有効または無効にするノードのアドレスを見つけます。
3. 左側のカラムで [Select] ボックスをチェックします。
4. 画面の下部で [**Delete**] をクリックします。
   確認画面が表示されます。
5. [**Delete**] をクリックします。

モニタの関連付けの削除

設定ユーティリティを使用して、特定のノードに明示的に関連付けられているモニタを削除できます。特定のノードに関連付けられているモニタを削除する場合、モニタの関連付けをすべて削除するか、デフォルトモニタのみがノードに関連付けられるように変更することが可能です。

また、デフォルトモニタ（作成するノードに BIG-IP によって自動的に関連付けられるモニタ）をすべて削除することもできます。

モニタの関連付けの詳細については、「*ヘルスモニタの割り当て*」（3-5 ページ）を参照してください。

ノードに対する明示的なモニタの関連付けを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Nodes**] をクリックします。
   [Nodes] 画面が開きます。
2. 管理するノードのアドレスをクリックします。
3. 画面の [Configuration] セクションで、[**Health Monitors**] 設定を見つけます。
4. [**Node Default**] または [**None**] を選択します。
5. [**Update**] をクリックします。

デフォルト モニタを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Nodes**] をクリックします。
   [Nodes] 画面が開きます。
2. [Default Monitor] メニューをクリックします。
3. [**Move**] ボタンを使用して（**>>**）、[**Active**] ボックス内のアクティブなモニタをすべて[**Available**]ボックス内に移動します。
4. [**Update**] をクリックします。

4

ロードバランシングプールの設定

- ロードバランシングプールの概要
- ロードバランシングプールの作成と変更
- プールの設定
- プールメンバの設定
- プールおよびプールメンバの管理

ロードバランシングプールの概要

典型的なクライアントサーバ構成のシナリオでは、クライアントリクエストは、リクエストヘッダで指定された宛先 IP アドレスに送信されます。大量の着信トラフィックがあるサイトでは、大量の要求を処理しようとする際に、宛先サーバがすぐに過負荷状態になる可能性があります。この問題を解決するために、BIG-IP® ローカルトラフィック管理システムでは、指定された宛先 IP アドレスだけでなく複数のサーバにクライアントリクエストを分散させます。ロードバランシングプールを作成する際に、これを行うように BIG-IP を設定します。

ロードバランシングプールについて

*ロードバランシングプール*とは、トラフィックを受信および処理するために同じグループにまとめられたデバイス（ Web サーバなど）の論理セット です。クライアントリクエスト で指定された宛先 IP アドレスにクライアントトラフィックを送信する代わりに、BIG-IP では、そのプールのメンバである任意のサーバにリクエストを送信します。これにより、サーバリソースの負荷を効率よく分散することができます。

プールを作成するとき、プールにプールメンバを割り当てます。*プールメンバ*は、ネットワーク上の物理ノード（サーバ）を表す論理オブジェクトです。さらに、プールを、BIG-IP 上のバーチャルサーバと関連付けます。プールをバーチャルサーバに割り当てると、BIG-IP は、バーチャルサーバに入るトラフィックをそのプールのメンバにダイレクトします。ネットワークトラフィックの管理方法に応じて、個々のプールメンバは、1 つまたは複数のプールに属します。

BIG-IP によってリクエストの送信先として選択される特定のプールメンバは、そのプールに割り当てているロードバランシング方式によって決定されます。*ロードバランシング方式*とは、要求を処理するプールメンバを選択するために使用されるアルゴリズムです。たとえば、デフォルトのロードバランシング方式は ***Round Robin*** です。この場合、BIG-IP は、次に使用可能なプールのメンバに着信要求のそれぞれを送信するため、要求はプール内のサーバ間で均等に分散されます。ロードバランシング方式の一覧については、「*ロードバランシング方式の指定*」（4-10 ページ）を参照してください。

ロードバランシングプールの機能

プールに対してさまざまな操作を実行するように、BIG-IP を設定できます。次の操作が可能です。

- ヘルスモニタをプールおよびプールメンバに関連付ける
- SNAT 接続を有効または無効にする
- 最初にターゲットにしたプールメンバが使用不可になった場合、別のプールメンバに接続を再バインドする
- パケット内で Quality of Service レベルまたは Type of Service レベルを設定する
- プールに対してロードバランシングアルゴリズムを指定する
- プール内のプライオリティグループにプールメンバを割り当てる

ロードバランシングプールの作成と変更

設定ユーティリティを使用して、ロードバランシングプールを作成したり、プールとそのメンバを変更します。プールを作成すると、BIG-IP によって、デフォルト設定のグループが、そのプールとそのメンバに自動的に割り当てられます。これらのデフォルト設定は、そのまま使用することも、変更することもできます。また、プールを作成した後でいつでも設定を変更できます。

BIG-IP の設定には、基本設定と詳細設定があります。プールの作成時にデフォルト設定のいくつかを変更する場合は、必ず画面上で [**Advanced**] オプションを選択して、構成可能な設定をすべて表示するようにしてください。基本設定および詳細設定の概要については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。

ロードバランシングプールの作成と実装

ロードバランシングプールの作成と実装は、次の 2 つの作業から成るプロセスです。

- まず、プールを作成します。
- 次に、そのプールをバーチャルサーバに関連付けます。

◆ 重要

*プールが作成されると、BIG-IP はそのプールを現在の管理パーティションに配置します。パーティションの詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

ロード バランシングプールを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Pools**] をクリックします。
   [Pools] 画面が開きます。
2. 画面右上の [**Create**] をクリックします。
   [New Pool] 画面が開きます。

   ***注**: [Create] ボタンをクリックできない場合は、現在のユーザロールではプールを作成する権限が与えられていないことを示します。*
3. [**Configuration**] リストから、[**Advanced**] を選択します。
4. [**Name**] 設定で、プールの名前を入力します。
5. その他の設定のそれぞれを指定するか、そのまま使用するか、または変更します。
   プールの設定については、「*プールの設定*」（4-6 ページ）またはこの画面のオンラインヘルプを参照してください。
6. [**Finished**] をクリックします。

ロード バランシングプールを実装するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Servers] 画面が開きます。
2. 適切なバーチャルサーバの名前をクリックします。
   選択したバーチャルサーバの設定が表示されます。
3. メニューバーで [**Resources**] をクリックします。
4. [**Default Pool**] リストで、新規作成したプールの名前を選択します。
5. [**Update**] をクリックします。

ロードバランシングプールの変更

既存のプールに対して行った任意の設定（ロードバランシング方式を含む）を変更できます。プールの設定については、「*プールの設定*」（4-6 ページ）またはオンラインヘルプを参照してください。既存のプールへのメンバの追加については、「*プールメンバシップの変更*」（4-3 ページ）を参照してください。

プールの設定を変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Pools**] をクリックします。
   [Pools] 画面が開きます。
2. 既存のプールの名前をクリックします。
   そのプールの既存の設定が表示されます。
3. [**Configuration**] リストから、[**Advanced**] を選択します。
   プールの設定が表示されます。
4. すべての設定を変更するか、またはそのまま使用します。
5. [**Update**] をクリックします。
6. ロードバランシング方式を変更するか、またはプライオリティグループのアクティベーションを有効または無効にする必要がある場合は、メニューバーで [**Members**] をクリックします。
7. [**Load Balancing Method**] および [**Priority Group Activation**] 設定を変更するか、そのまま使用します。
8. [**Update**] をクリックします。

プールメンバシップの変更

既存のロードバランシングプールについては、既存のプールメンバを変更するか、または新規メンバをプールに追加することができます。

既存のプールメンバの変更

プールのメンバの設定を変更する場合、次のことを行えます。

- プールメンバを有効または無効にする
- プールからメンバを削除する
- プールメンバの設定値を変更する

既存のプールメンバを変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Pools**] をクリックします。
   [Pools] 画面が開きます。
2. [Members] カラムで、表示された番号をクリックします。
   そのプールの既存のメンバがリストされます。
3. 画面の [Current Members] セクションを見つけます。
4. プールメンバを変更します。
   a) プールメンバを有効または無効にするか、またはプールからメンバを削除する場合は、メンバアドレスの左側にあるボックスをクリックします。次に、[**Enable**]、[**Disable**]、または [**Remove**] をクリックします。
   b) プールメンバの設定を変更する必要がある場合は、アドレスをクリックし、必要に応じてプールメンバの設定をそのまま使用するか変更します。プールメンバの設定については、「*プールメンバの設定*」（4-16 ページ）を参照してください。
5. [**Update**] をクリックします。

既存のロードバランシングプールへのメンバの追加

プールの作成時にプールメンバを指定できるだけでなく、後でプールメンバを追加することも可能です。（プールの作成時にプールメンバを指定するのではなく）既存のプールにプールメンバを追加する場合は、そのプールメンバに対してさまざまな設定を行えます。明示的に指定する必要がある設定は、[**Address**] および [**Service Port**] 設定のみです。その他の設定にはすべて、デフォルト値があります。この値は、必要に応じて、そのまま使用することも調整することも可能です。

◆注

*プールの作成時にプールメンバを指定する場合は、プールメンバの設定は表示されず、BIG-IP によってデフォルト値が割り当てられるだけです。ただし、後でプールメンバのプロパティを変更して設定を調整することが可能です。詳細については、「**プールおよびプールメンバの管理**」（4-20 ページ）を参照してください。*

ロード バランシングプールにメンバを追加するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Pools**] をクリックします。
   [Pools] 画面が開きます。

2. [Members] カラムで、表示された番号をクリックします。
そのプールの既存のメンバがリストされます。
3. 画面の右側で [**Add**] をクリックします。
[New Pool Members] 画面が開きます。
4. [**Address**] ボックスで、[**New Address**] を選択して IP アドレスを入力するか、[**Node List**] を選択して IP アドレスを選択します。
5. [**Service Port**] ボックスで、ポート番号を入力するか、リストからサービスを選択します。
6. その他の設定をすべてそのまま使用するか、設定します。プールメンバの設定については、「*プールメンバの設定*」（4-16 ページ）を参照してください。
7. [**Finished**] をクリックします。

プールの設定

プールの設定を行って、必要に応じてプールを調整できます。デフォルト値を持つ設定については、デフォルト設定をそのまま使用することも変更することもできます。また、設定の変更は、プールの作成時に行うか、プールの作成後にいつでも行うことができます。設定ユーティリティを使用してこれらの設定を行う方法については、「*ロードバランシングプールの作成と変更*」(4-2 ページ) を参照してください。

表 4.1 に、プールに対して行える設定を列挙し、それぞれの設定について説明します。

プール設定	説明	デフォルト値
Name	プールのユーザ定義名を指定できます。プール名前の指定は必須です。	デフォルト値なし
Health Monitors	ヘルスモニタまたはパフォーマンスモニタを、個々のプールメンバだけではなくプール全体に関連付けることができます。これにより、複数の Web サーバに対して健全性およびパフォーマンスの監視を設定する作業が容易になります。	デフォルト値なし
Availability Requirement	プールメンバが **up** 状態として定義される前に、そのメンバを利用可能であるとしてレポートする必要があるモニタの数を指定できます。	**All**
Allow SNAT	特定のプールを使用するすべての接続に対して SNAT が自動的に有効または無効にされるようにプールを設定できます。	**Yes**
Allow NAT	特定のプールを使用するすべての接続に対して NAT が自動的に有効または無効にされるようにプールを設定できます。	**Yes**
Action on Service Down	この設定が有効になっており、ターゲットのプールメンバがダウンした場合、BIG-IP によって別のプールメンバを選択し、クライアント接続を新しいサーバ接続に再バインドすることが試行されます。指定できる値は、[**None**]、[**Reject**]、[**Drop**]、および [**Reselect**] です。	**None**
Slow Ramp Time	このオプションを選択した場合、BIG-IP によって、通常より少ない量のトラフィックが、新たに有効にされたプールメンバに、指定された時間だけ送信されます。	**0**
IP ToS to Client	ターゲットのプールに基づいて、クライアントへ送られるパケット内で特定の Type of Service（ToS）レベルを設定するようにプールを設定できます。	**Pass Through**
IP ToS to Server	ターゲットのプールに基づいて、サーバへ送られるパケット内で特定の Type of Service（ToS）レベルを設定するようにプールを設定できます。	**Pass Through**
Link QoS to Client	ターゲットのプールに基づいて、クライアントへ送られるパケット内で特定の Quality of Service（QoS）レベルを設定するようにプールを設定できます。	**Pass Through**
Link QoS to Server	ターゲットのプールに基づいて、サーバへ送られるパケット内で特定の Quality of Service（QoS）レベルを設定するようにプールを設定できます。	**Pass Through**

表 4.1 *ロードバランシングプールの設定*

プール設定	説明	デフォルト値
Load Balancing Method	デフォルトのロードバランシング方式を使用するか、または別のロードバランシング方式を定義して、プライオリティベースのメンバのアクティベーションを設定できます。さまざまなロードバランシング方式を使用して、さまざまなプールを設定できます。	**Round Robin**
Priority Group Activation	プール内のプライオリティグループにプールメンバを割り当てることができます。	**Disabled**
New Members	作成するプールごとに、そのプールのメンバにするサーバを指定する必要があります。プールメンバは、その IP アドレスによって指定する必要があります。プールメンバごとに、サービスポート、重み付け、およびプライオリティグループを割り当てることもできます。	デフォルト値なし

表 4.1 *ロードバランシングプールの設定*

プールを設定する前に、変更する可能性がある特定のプール設定に関する説明を読んでおくと役立ちます。

プール名の指定

プールに対して行える最も基本的な設定は、プール名です。プール名では大文字小文字が区別され、文字、数字、および下線（_）のみを含めることができます。予約済みキーワードは使用できません。

定義するプールごとに一意の名前が必要です。

ヘルスモニタとプールとの関連付け

モニタは、BIG-IP の主要な機能です。モニタを使用すると、サーバが **up** 状態でトラフィック受信が可能であることを確認できます。モニタをサーバのプール全体に関連付ける場合、そのモニタを個々のサーバに明示的に関連付ける必要はありません。その場合は、プール設定 **Health Monitors** を使用して、モニタをプール自体に割り当てるだけです。これにより、BIG-IP が各プールメンバを自動的に監視するようになります。

BIG-IP には、監視対象のトラフィックタイプに応じて、プールに関連付けることができる各種の設定済みモニタが含まれています。独自のカスタムモニタを作成してプールに関連付けることも可能です。プールに関連付けることができない唯一の設定済みモニタタイプは、プールやプールメンバではなくノードを監視するために特に設計されたモニタです。すなわち、モニタの宛先アドレスには、IP アドレスとサービスポートではなく、IP アドレスのみを指定します。これに該当するモニタタイプは次のとおりです。

- ICMP
- TCP Echo
- Real Server
- SNMP DCA
- SNMP DCA Base
- WMI

BIG-IP では、さまざまな方法でモニタの関連付けを設定できます。

- 個々のサーバではなくプール全体にモニタを関連付けることができます。その場合、BIG-IP により、そのモニタがすべてのプールメンバ（後で追加したものを含む）に自動的に関連付けられます。同様に、プールからメンバを削除した場合、BIG-IP によってそのサーバは監視されなくなります。
- プールメンバとして指定されているサーバによって複数のプロセスが同じ IP アドレスおよびポートに存在することが許可されている場合は、各プロセスの健全性またはステータスをチェックできます。これを行うには、サーバを複数のプールに追加した後、各プール内で、モニタをそのサーバに関連付けます。各サーバに関連付けるモニタは、そのサーバで実行中のプロセスの健全性またはパフォーマンスをチェックします。
- モニタをプール全体に関連付ける場合は、個々のプールメンバをそのモニタへの関連付けから除外できます。その場合、その特定のプールメンバに対して別のモニタを関連付けるか、またはそのプールメンバを健全性の監視全体から除外できます。たとえば、プールメンバ A、B、および D を **http** モニタに関連付け、プールメンバ C を **https** モニタに関連付けることができます。
- 複数のモニタを同じプールに関連付けることができます。たとえば、**http** モニタと **https** モニタの両方を同じプールに関連付けることが可能です。

ヘルスモニタおよびパフォーマンスモニタの詳細については、第 12 章「*モニタの設定*」を参照してください。

アベイラビリティ要件の指定

この設定は、ヘルスモニタの最小数を指定します。BIG-IP によってプールメンバが **up** 状態としてレポートされる前に、最低でもこの数のモニタがプールメンバをトラフィック受信可能としてレポートする必要があります。

この設定を行うには、[**Availability Requirement**] ボックス内に数値を入力します。

SNAT および NAT の許可

プールを設定する場合、そのプールを使用するすべての接続に対して、セキュアネットワークアドレス変換（SNAT）またはネットワークアドレス変換（NAT）を無効にすることができます。これを行うには、[**Allow SNAT**] および [**Allow NAT**] を設定します。デフォルトでは、これらの設定は有効になっています。既存のプールでこの設定を変更するには、そのプールの [Properties] 画面を表示します。

SNAT 接続または NAT 接続を無効にするようにプールを設定する必要がある状況の例としては、特定のサービスに対してプールにおける SNAT 接続または NAT 接続を無効にする場合があります。この場合は、そのサービスに対してすべての接続を処理する別個のプールを作成した後、そのプールに対して SNAT または NAT を無効にします。

SNAT および NAT の概要については、第 14 章「*SNAT の設定*」を参照してください。

サービスが利用不可になったときの処理の指定

[**Action on Service Down**] 設定では、プールメンバ上のサービスが利用不可になったときに BIG-IP で行う処理を指定します。可能な設定は次のとおりです。

- **None** -- BIG-IP は、アクションを実行しません。これはデフォルトの処理です。
- **Reject** -- BIG-IP は、RST（TCP のみ）または ICMP メッセージを送信します。
- **Drop** -- BIG-IP は、接続をクリーンアップします。
- **Reselect** -- BIG-IP は、別のノードを選択します。

この設定を行うには、[**Action on Service Down**] 設定で、リストから値を選択します。

スローランプタイムの設定

プールメンバをオフラインにしてからオンラインに戻した場合、プールに使用するロードバランシングモードによっては、プールメンバが接続要求で過負荷状態になる可能性があります。たとえば、Least Connections ロードバランシングモードを使用する場合、BIG-IP によって新しい接続はすべて、新たに有効にされたプールメンバに送信されます（理論上、そのメンバに最小量の接続があるため）。

[**Slow Ramp Time**] を設定すると、新たに有効にされたプールメンバに送信されるトラフィックの量が少なくなります。トラフィックの量は、プールメンバが利用可能になっている時間とスローランプタイムとの比率（秒単位）に基づいています。プールメンバがオンラインになっている時間がスローランプタイムを上回ると、プールメンバは着信トラフィック全体を受信します。

この設定を行うには、[**Slow Ramp Time**] 設定で数値を入力します。この数値は、トラフィックを新しく有効にしたプールメンバへ送る前に BIG-IP が待機する秒数を表します。

Type of Service（ToS）レベルの設定

もう 1 つのプール設定に、Type of Service（ToS）レベルがあります。***ToS*** レベルは、ネットワーク機器が識別子に基づいて異なる方法でトラフィックを識別および処理できる手段となります。

トラフィックがサイトに入ると、BIG-IP は、パケットに ToS レベルを設定できます。パケットの送信先となるプールに定義した **IP ToS to Server** ToS レベルを使用して、BIG-IP は、iRule を適用し、該当する ToS レベルに基づいて、トラフィックをサーバの別のプールへ送ることができます。

また、BIG-IP では、プールで設定された **IP ToS to Client** ToS 値に基づいて、発信トラフィック（すなわち、**HTTP GET** に基づいたリターンパケット）にタグ付けすることができます。その後、この値はアップストリームデバイスによって検査され、適切なプライオリティが与えられます。

たとえば、特定のプールに送信されるパケットに対して ToS レベルが設定されるようにプールを設定するには、[**IP ToS to Client**] レベルと [**IP ToS to Server**] レベルの両方を **16** に設定します。この場合、クライアントにパケットを送信するときも、サーバにパケットを送信するときも、ToS レベルが **16** に設定されます。

◆注

*クライアントまたはサーバのプールで **ToS** レベルを変更した場合、既存の接続では以前の設定が引き続き使用されます。*

Quality of Service（QoS）レベルの設定

もう 1 つのプールの設定として、Quality of Service（QoS）レベルがあります。ToS レベルに加えて、***QoS*** レベルもまた、ネットワーク機器が識別子に基づいて異なる方法でトラフィックを識別および処理できる手段となります。基本的に、パケットで指定された QoS レベルによって、そのパケットのスループットポリシーが実施されます。

トラフィックがサイトに入ると、BIG-IP は、パケットに QoS レベルを設定できます。パケットの送信先となるプールに定義した **IP QoS to Server** QoS レベルを使用して、BIG-IP は、該当する ToS レベルに基づいて、トラフィックをサーバの別のプールへ送る iRule を適用できます。

また、BIG-IP では、プールで設定された **IP QoS to Client** QoS 値に基づいて、発信トラフィック（すなわち、**HTTP GET** に基づいたリターンパケット）にタグ付けすることができます。その後、この値はアップストリームデバイスによって検査され、適切なプライオリティが与えられます。

たとえば、特定のプールに送信されるパケットに対して QoS レベルが設定されるようにプールを設定するには、[**Link QoS to Client**] レベルを **3** に設定し、[**Link QoS to Server**] レベルを **4** に設定します。この場合、クライアントにパケットを送信するときは QoS レベルが **3** に設定され、サーバにパケットを送信するときは **4** に設定されます。

ロードバランシング方式の指定

ロード バランシングは、BIG-IP の不可欠な部分です。BIG-IP でロード バランシングを設定することは、ロード バランシング シナリ オ、つまり 、どのプールメンバが特定のバーチャルサーバによってホストされる接続を受信するかを決定することを意味します。ロード バランシング シナリ オを決定した場合は、そのシナリ オに適したロード バランシング 方式を指定できます。

*ロードバランシング方式*は、BIG-IP によってトラフィックの送信先とするノードを決定するために使用されるアルゴリズムまたは公式です。個々のロードバランシング方式では、現在の接続カウントなど、1 つ以上のダイナミック因子を考慮に入れます。BIG-IP の用途はそれぞれ固有であり、ノードのパフォーマンスは多数の異なる因子によって左右されます。したがって、各種のロードバランシング方式を試行して、各自の環境で最適なパフォーマンスが得られる方式を選択することをお勧めします。

デフォルトのロードバランシング方式の使用

BIG-IP のデフォルトのロードバランシング方式は、Round Robin です。この方式では、新しい接続要求のそれぞれをリスト内の次のサーバに渡すだけです。その他すべてのロードバランシング方式では、サーバのキャパシティやステータスを考慮に入れます。

ロードバランス対象機器の処理速度とメモリがほぼ等しい場合、Round Robin モードはほとんどの構成において有効です。Round Robin 方式を使用する場合は、この項の残りの部分を飛ばして、基本プール構成に追加するその他のプール設定を始めてもかまいません。

ロードバランシング方式の選択

複数の異なるロードバランシング方式から選択できます。処理速度とメモリが大きく異なるサーバで作業する場合には、Ratio または Dynamic Ratio 方式のいずれかに切り替えることができます。

◆注

一部のハードウェアプラットフォームでは、さらに別のロードバランシング方式を利用できる場合があります。

- **Round Robin**
  これはデフォルトのロードバランシング方式です。Round Robin モードでは、新しい接続要求のそれぞれをリスト内の次のサーバに渡すことで、ロードバランス対象マシンの配列間で接続を均等に分散させます。ロードバランス対象機器の処理速度とメモリがほぼ等しい場合は特に、Round Robin モードはほとんどの構成において有効です。

- **Ratio(member)** および **Ratio(node)**
  BIG-IP では、定義した重み付けに従ってマシン間で接続を分散させます。各マシンで経時的に受信される接続数は、各マシンに対して定義した重み付けに比例しています。これらはスタティックロードバランシング方式であり、サーバのキャパシティに比例したスタティックなユーザ割り当ての重み付けに基づいて分散が行われます。Ratio ロードバランシングとは以下のようなものです。

  ロードバランシング計算は、各プールに限定するか（メンバベースの計算）、またはサーバがメンバであるすべてのプールに適用できます（ノードベースの計算）。この区別は、Ratio 方式では特に重要です。Ratio（Member）方式では、実際の重み付けは、プール定義におけるメンバ設定ですが、Ratio（node）方式では、重み付けはノードの設定です。

すべてのノードについて、デフォルトの比率設定は **1** です。Ratio（member）ではなく Ratio（node）ロードバランシング方式を使用する場合は、構成内の 1 つ以上のノードに対して **1** 以外の比率を設定する必要があります。少なくとも 1 つの比率設定を変更しない場合、ロードバランシング方式には、Round Robin ロードバランシング方式と同じ効果があります。

***警告**：ロードバランシング方式を Ratio（member）ではなく Ratio（node）に設定する場合、各ノードに対して比率設定を定義する必要があります。*

- **Dynamic Ratio (member) および Dynamic Ratio (node)**
  Dynamic Ratio 方式は、Ratio 方式と 同様です。ただし 、重み付けはサーバの継続的な監視に基づいているため、絶えず変化しています。

  これはダイナミックロードバランシング方式であり、リアルタイムサーバパフォーマンス分析のさまざまな側面（ノードごとの現在の接続数、Fastest ノード応答時間など）に基づいて接続を分散させます。

  Dynamic Ratio方式は、特にRealNetworks® RealSystem® Serverプラットフォーム、Windows Management Instrumentation（WMI）が搭載された Windows® プラットフォーム、または SNMP エージェント（UC Davis SNMP エージェント、Windows 2000 Server SNMP エージェントなど）が搭載された任意のサーバへのトラフィックをロードバランスするために使用されます。Dynamic Ratio ロードバランシングを実装するには、まずこれらのシステムに必要なサーバソフトウェアをインストールして設定した後、適切なパフォーマンスモニタをインストールします。詳細については、付録 B「*モニタについての追加留意事項*」を参照してください。

- **Fastest（node）**および **Fastest（application）**
  Fastest 方式では、プールメンバに対する未処理のレイヤ 7 要求の最少数、およびオープンなレイヤ 4 接続の数に基づき、新しい接続を渡します。BIG-IP がレイヤ 7 要求を受信すると、その接続を処理しているプールメンバのカウンタが増加し、対応するレイヤ7応答をノードから受信した時点でそのカウンタが減少します。このカウンタは、ノードに示された要求のうち、現時点でまだ応答を受け取っていない要求の数を示します。Fastest 方式は、ノードが異なる論理ネットワーク間に分散している環境で特に有用だと考えられます。ロードバランシング計算は、各プールに限定するか（メンバベースの計算）、またはサーバがメンバであるすべてのプールに適用できます（ノードベースの計算）。

  Fastestロードバランシング方式には以下のルールが適用されます。

  - レイヤ 7 プロファイルと TCP プロファイルの両方をバーチャルサーバに割り当てる必要があります。
  - レイヤ 7 プロファイルを設定しなければ、バーチャルサーバでは Least Connections ロードバランシングモードが使用されることになります。

- **Least Connections (member)** および **Least Connections (node)**
  Least Connections 方式は比較的単純な方式であり、BIG-IP は新しい接続を、現在の接続数が最小であるノードに渡します。

Least Connections 方式は、ロードバランス対象のサーバまたはその他の機器に同様の機能がある環境において最も有効です。これらは動的なロードバランシング方式です。

ロードバランシング計算は、各プールに限定するか（メンバベースの計算）、またはサーバがメンバであるすべてのプールに適用できます（ノードベースの計算）。

- **Observed（member）**および **Observed（node）**
  Observed 方式では、接続数に基づきノードがランク付けされ、最小接続数のバランスのよいノードほど、より多くの接続を受信します。

  Observed 方式と Least Connections 方式の違いは、Least Connections 方式がロードバランシング実行時にしか接続を測定しないのに対して、Observed 方式は各ノードに対するレイヤ 4 接続の数を継続的に追跡し、ロードバランシング比率を算出する点にあります。

  Observed モードはあらゆる環境で有効に機能しますが、ノードのパフォーマンスが大きく変動する環境で特に有用だと考えられます。これらは動的なロードバランシング方式です。

  ロードバランシング計算は、各プールに限定するか（メンバベースの計算）、またはサーバがメンバであるすべてのプールに適用できます（ノードベースの計算）。

- **Predictive（member）**および **Predictive（node）**
  Predictive 方式でも、Observed 方式と同じランキング方式が用いられ、現在の接続数に従いノードがランク付けされます。ただし、Predictive 方式では、BIG-IP は経時的なランキングの傾向を分析して、ノードのパフォーマンスが現在向上しているか劣化しているかを判断します。より優れた（現在劣化しているのではなく向上している）パフォーマンスランキングを持つノードが、より高い比率の接続を受信します。Predictive 方式は、あらゆる環境で有効です。これらは動的なロードバランシング方式です。

  ロードバランシング計算は、各プールに限定するか（メンバベースの計算）、またはサーバがメンバであるすべてのプールに適用できます（ノードベースの計算）。

- **Least Sessions**
  Least Sessions 方式では、現時点でのパーシステントセッション数が最小であるノードに新しい接続を渡します。このロードバランシング方式を使用するには、パーシステンス接続を追跡するパーシステンスプロファイルをバーチャルサーバが参照する必要があります。このようなパーシステンスプロファイルには、Source Address Affinity や Universal などがあります。Least Sessions 方式は、ロードバランシングの対象となるサーバなどの機器が同様の機能を持つ環境において、もっとも有効に機能します。これは動的なロードバランシング方式です。

- **L3 Address**
  L3 Address 方式では、メンバーの IP アドレスを使って、各メンバーに連続して接続を渡します この IP アドレスはレイヤ 3（L3）アドレスです。

プライオリティベースのメンバのアクティベーションの指定

トラフィックのロードバランスは、プールのすべてのメンバ間で行うことも、プライオリティ値に従って現在アクティブになっているメンバ間のみで行うことも可能です。プライオリティベースのメンバのアクティベーションでは、プール内の各メンバには、プライオリティ値が割り当てられます。このプライオリティ値によって、プール内の各メンバは、その数で指定されたプライオリティグループ内に配置されます。

すべてのプールメンバが利用可能である（つまり、有効になっており、**up** としてマークされ、コネクションリミットを超えていない）場合、BIG-IP によって、最高プライオリティのグループ（つまり、最高のプライオリティ値によって指定されたグループ）のみのメンバすべてに接続が分散されます。**Priority Group Activation** の値によって、そのグループに限定されるトラフィックに対して利用可能のままにする必要がある最小メンバ数が決定されます。最高プライオリティのグループで利用可能なメンバの数が最小数を下回った場合、BIG-IP によってトラフィックは、2 番目に高いプライオリティのグループにも分散されます（以降も同様に行われます）。図 4.1（4-14 ページ）に示した設定では、3 つのプライオリティグループ、**3**、**2**、および **1** があります。

```
pool my_pool {
    lb_mode fastest
    min active members 2
    member 10.12.10.7:80 priority 3
    member 10.12.10.8:80 priority 3
    member 10.12.10.9:80 priority 3
    member 10.12.10.4:80 priority 2
    member 10.12.10.5:80 priority 2
    member 10.12.10.6:80 priority 2
    member 10.12.10.1:80 priority 1
    member 10.12.10.2:80 priority 1
    member 10.12.10.3:80 priority 1
    }
```

***図 4.1** プライオリティロードバランシングのサンプルプール設定*

接続はまず、プライオリティ **3**（最高プライオリティのグループ）のすべてのプールメンバに分散されます。利用可能なプライオリティ **3** のメンバ数が 2 より少ない場合、トラフィックはプライオリティ **2** のメンバにもダイレクトされます。プライオリティ **3** グループとプライオリティ **2** グループのいずれでも、利用できるメンバが 2 未満の場合、トラフィックは、プライオリティ **1** グループへダイレクトされます。このとき、BIG-IP は、高いプライオリティのグループを継続して監視し、プライオリティの高いグループで利用可能な最小メンバ数になるたびに、そのグループへのトラフィックを制限します。

プールメンバの指定

この設定を行う際は、ロードバランシングプールを構成するサーバ（つまり、プールメンバ）を指定します。プールメンバを指定するには、[**Address**] および [**Service Port**] 設定を使用して、サーバの IP アドレスおよびサービスポートを指定する必要があります。

オプションの設定に重み付けがあります。この設定は、**Ratio (member)**、**Ratio (node)**、または **Dynamic Ratio** のロードバランシング方式と、プライオリティグループのアクティベーションを選択している場合に適用できます。これらの設定の詳細については、次の「*プールメンバの設定*」を参照してください。

プールメンバの設定

プールにメンバを追加する場合は、そのプールメンバに対してさまざまな設定を行えます。これらの設定のほとんどは、ロードバランシングプールの作成後に行います。プールの作成時に指定する必要がある設定は、[**Address**] および [**Service Port**] 設定のみです。その他の設定にはすべて、デフォルト値があります。この値は、必要に応じて、そのまま使用することも、後で調整することも可能です。

プール作成時のプールメンバの追加については、「*ロードバランシングプールの作成と実装*」（4-2 ページ）を参照してください。既存のプールへのメンバの追加については、「*既存のロードバランシングプールへのメンバの追加*」（4-4 ページ）を参照してください。

◆注

プールとそのメンバは、常に同じパーティションに置かれるようになっています。

表 4.2 に、プールメンバに対して行える設定を示し、それぞれの設定について説明します。

汎用プロパティ	説明	デフォルト値
Ratio	プールメンバに割り当てる重み付けを指定します。	**1**
Priority Group	プールメンバのプライオリティグループを指定します。	**1**
Connection Limit	プールメンバに許可された同時接続の最大数を指定します。	**0**
Health Monitors	プールメンバがプールに関連付けられたモニタを継承するか、別のモニタを使用するかを指定します。	**Inherit From Pool**
Select Monitors	対象のプールメンバに関連付ける1つまたは複数のモニタを指定します。この設定は、[**Health Monitors**] を [**Member Specific**] に設定した場合にのみ使用します。	デフォルト値なし
Availability Requirement	ヘルスモニタの最小数を指定します。BIG-IPによってプールメンバが **up** 状態としてレポートされる前に、最低でもこの数のモニタがプールメンバをトラフィック受信可能としてレポートする必要があります。	

***表 4.2** 個々のプールメンバの構成設定*

プールメンバをプールに追加した後でプールメンバの設定を調整するには、「*プールメンバプロパティを表示するには*」（4-24 ページ）を参照してください。

プールメンバを追加する前に、変更する可能性がある特定のプールメンバの設定に関する説明を読んでおくと役立ちます。

プールメンバの重み付けの指定

比率ベースのロードバランシング方式を使用してプール内のサーバにトラフィックを分散させる場合は、[**Ratio**] 設定を使用してサーバに重み付けを割り当てることができます。重み付けによって、サーバが受信するトラフィックの量が決まります。

重み付けロードバランシング方式は、**Ratio (member)**、**Ratio (node)**、および **DynamicRatio** です。比率ベースのロードバランシング方式の詳細については、「*ロードバランシング方式の指定*」(4-10 ページ) および付録 B「*モニタについての追加留意事項*」を参照してください。

プライオリティベースのメンバのアクティベーションの指定

[**Priority**] 設定では、プライオリティ値をプールメンバに割り当てます。その後、プール内で、プールメンバに割り当てられたプライオリティ値に従ってトラフィックがロードバランスされます。したがって、高いプライオリティが割り当てられているメンバは、負荷が特定のレベルに達するまでトラフィックを受信します。その後、トラフィックは、次に高いプライオリティを持つグループに割り当てられているメンバにダイレクトされます。

[**Priority Group Activation**] 設定を使用すると、BIG-IP がより低いプライオリティのメンバにトラフィックをダイレクトし始めるタイミングを決定する負荷レベルを設定できます。詳細については、「*プライオリティベースのメンバのアクティベーションの指定*」(4-14 ページ) を参照してください。

コネクションリミットの指定

[**Connection Limit**] 設定では、プールメンバに許可されている同時接続の最大数を指定できます。デフォルト値 **0** (ゼロ) を指定した場合、プールメンバで受信可能な同時接続の数に制限はありません。

明示的なモニタ関連付けの選択

モニタをプールに関連付けた場合、BIG-IP によってそのモニタが各プールメンバ (後でプールに追加するメンバを含む) に自動的に関連付けられます。ただし、場合によっては、プールに割り当てられているモニタとは別のモニタを、特定のプールメンバに割り当てる必要があります。その場合は、[**Health Monitors**] 設定を使用して、特定のモニタをプールメンバに明示的に割り当てるように指定する必要があります。

また、この設定を行うことで、BIG-IP によって特定のプールメンバにモニタが関連付けられないようにすることも可能です。

モニタをプールメンバに明示的に関連付けるには、[**Health Monitors**] 設定で [**Member Specific**] を選択することで、[**Select Monitors**] 設定を表示します。さらに、次の項の説明に従って [**Select Monitors**] の設定を行います。

BIG-IP によってプールメンバにモニタが関連付けられないようにするには、[**Health Monitors**] を [**None**] に設定します。

プールメンバに対する明示的なモニタの関連付けの作成

BIG-IP には、監視対象のトラフィックタイプに応じて、プールメンバに関連付けることができる各種のモニタが含まれています。独自のカスタムモニタを作成してプールメンバに関連付けることも可能です。プールメンバに関連付けることができない唯一のモニタタイプは、プールやプールメンバではなくノードを監視するために特に設計されたモニタです。これに該当するモニタタイプは次のとおりです。

- ICMP
- TCP Echo
- Real Server
- SNMP DCA
- SNMP DCA Base
- WMI

ヘルスモニタおよびパフォーマンスモニタの詳細については、第 12 章「*モニタの設定*」を参照してください。

モニタを個々のプールメンバに関連付けるには、プールメンバの設定を表示して、[**Health Monitors**] 設定を [**Member Specific**] に設定します。[**Select Monitors**] 設定が表示されます。プールメンバに関連付けるモニタを選択し、左矢印（**<<**）を使用して、モニタ名を [**Active**] ボックスに移動します。[**Finished**] または [**Update**] をクリックして、選択したプールメンバのみに対してモニタの関連付けをアクティブにします。

複数のモニタを同じプールメンバに関連付ける

BIG-IP では、1 つ以上のモニタを同じサーバに関連付けることができます。設定ユーティリティを使用して、次のことを行えます。

- **複数のモニタを単一プールのメンバに関連付ける。**
  たとえば、モニタ **http1**、**http2**、および **http3** を作成し、各モニタを異なる設定にして、3 つのモニタすべてを同じプールメンバに関連付けることができます。この場合、いずれかのチェックに失敗すると、そのプールメンバは **down** としてマークされます。
- **1 つの IP アドレスおよびサービスを、複数のプールのメンバとして割り当てる。**
  この場合は、各プール内で、そのプールメンバに別のモニタを割り当てることができます。たとえば、サーバ **10.10.10:80** を、3 つの別個のプール、**my_pool1**、**my_pool2**、および **my_pool3** に割り当てるとします。この場合、3 つのカスタム HTTP モニタ（プールごとに 1 つのモニタ）をすべて同じサーバに関連付けることができます。その結果、BIG-IP では、**http1** モニタを使用してプール **my_pool1** 内のサーバ **10.10.10.:80** の健全性をチェックし、**http2** モニタを使用してプール **my_pool2** 内のサーバ **10.10.10.:80** の健全性をチェックし、**http3** モニタを使用してプール **my_pool3** 内のサーバ **10.10.10:80** の健全性をチェックするようになります。

複数モニタの関連付けは、プールメンバを各プールに追加する際、または後でプールメンバのプロパティを変更する際に行えます。

アベイラビリティ要件の指定

[**Availability Requirement**] 設定では、ヘルスモニタの最小数を指定します。BIG-IP によってプールメンバが **up** 状態としてレポートされる前に、最低でもこの数のモニタがプールメンバをトラフィック受信可能としてレポートする必要があります。

プールおよびプールメンバの管理

一般に、プールおよびプールメンバを管理する際は、既存のプールまたはプールメンバの設定を表示する必要があります。場合によっては、その他の管理作業も行う必要があります。設定ユーティリティを使用して、次のことを行えます。

- プールを管理する
- プールメンバを管理する
- プールおよびプールメンバのモニタ関連付けを無効にする
- プールおよびプールメンバの統計情報を表示する

プールとプールメンバの管理で重要なのが、プールまたはプールメンバのステータスをいつでも表示し、把握できることです。設定ユーティリティでは、複数のアイコンのいずれかを表示して、ステータスを示します。これらのアイコンは、各プールやプールメンバの色や形で区別できます。

- アイコンの*形*は、プールまたはプールメンバでモニタからレポートされたステータスを示します。たとえば、丸形のアイコンは、プールメンバが **up** であることがモニタで報告されたことを示し、ダイヤモンド形のアイコンは、プールメンバが **down** であることがモニタで報告されたことを示します。
- アイコンの*色*は、ノードの実際のステータスを示します。たとえば、緑のアイコンは、ノードが **up** であることを示し、赤いアイコンは、ノードが **down** であることを示します。また、黒いアイコンは、ユーザの介入が必要であることを示します。

ステータスアイコンの詳細については、「*プールステータスについて*」（4-22 ページ）と「*プールメンバステータスについて*」（4-24 ページ）を参照してください。

プールプロパティの変更については、「*ロードバランシングプールの変更*」（4-3 ページ）を参照してください。プールメンバプロパティの変更については、「*既存のプールメンバの変更*」（4-4 ページ）を参照してください。

プールの管理

BIG-IP では、既存のロード バランシングプールを維持するため、いくつかのプール固有のタスクを実行できます。管理権限を持つプールに対して、ユーザはプール一覧の確認、プールプロパティの表示、プールステータスの確認、プールの削除を行えるようになっています。

◆注

ユーザロールとパーティションアクセス割り当てに基づき、管理権限を持つプールのみを管理することができます。

プールのリストの表示

表示が許可されている既存のプールの一覧を表示することができます。プールのリストを表示する場合、設定ユーティリティによって、プールに関する次の情報が表示されます。

- 状態
- 名前
- プールが配置されたパーティション
- プールメンバ数

BIG-IP で定義されたプールのリストを表示するには、次の手順に従います。

プールのリストを表示するには

ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Pools**] をクリックします。
[Pools] 画面が開き、表示権限のあるプールの一覧が表示されます。

プールプロパティの表示

プールの汎用プロパティを表示するには、設定ユーティリティを使用します。ただし、表示できるのは、表示権限のあるプールのプロパティにかぎられます。これらのプールプロパティとその説明を以下に示します。

- **名前**
  プールに割り当てた一意の名前。プール名の例としては、**my_http_pool** があります。
- **パーティション**
  プールが配置された管理パーティション。管理パーティションの詳細については、『***TMOS™ Management Guide for BIG-IP® Systems***』を参照してください。
- **アベイラビリティ**
  以下に基づいた、プールのステータス。
  - プールが有効になっているかどうか
  - 親ノードのステータス
  - 関連付けられたヘルスモニタからのレポートに基づいた、プールメンバのステータス

プールプロパティを表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Pools**] をクリックします。
   [Pools] 画面が開きます。
2. プール名をクリックします。
   選択したプールのプロパティが表示されます。

プールステータスについて

プールのステータスは、いつでも確認できます。プールのステータスは、対応するメンバのステータスのみに基づきます。設定ユーティリティを使用すると、プールの **Availability** プロパティを表示して、この情報を確認できます。また、この情報は、プールのリストを表示し、[Status] カラムを確認することによっても把握できます。

設定ユーティリティでは、複数のアイコンのいずれかを表示して、プールのステータスを示します。これらのアイコンは、色や形で区別できます。アイコンについては、表 4.3 を参照してください。設定ユーティリティ内でアイコンを見つけるには、このページの「*プールプロパティを表示するには*」（4-21 ページ）を参照してください。ステータスアイコンの背景情報については、「*プールおよびプールメンバの管理*」（4-20 ページ）を参照してください。

ステータスインジケータ	説明
	トラフィックの処理に、少なくとも 1 つ以上のプールメンバが利用できます。
	現在、プールメンバは利用できませんが、ユーザが処理を行わなくても、後でいずれかが利用可能になります。使用できないプールメンバが自動的に使用可能になる例としては、プールメンバへの同時接続数が、プールメンバに定義された **[Connection Limit]** 設定の値を下回った場合があげられます。
	すべてのプールメンバが利用できないため、トラフィックを受け入れられません。プールメンバが利用できない理由としては、関連付けられた EAV モニタで、プールメンバの利用不可が検知されたことがあげられます。プールステータスが赤色である場合、通常は、ユーザの処理が必要となります。
	少なくとも 1 つ以上のプールメンバのステータスが不明であり、その他のプールメンバは利用できません。不明のプールメンバステータスには、以下のような原因があります。 1 つ以上のプールメンバにモニタが関連付けられていない。 モニタの結果がまだ出力されていない。 プールメンバの IP アドレスの設定が誤っている。 親ノードがネットワークから切断された。

表 4.3 *プールのステータスインジケータの説明*

プールの削除

既存のプールを削除するには、次の手順に従います。プールからの個々のプールメンバの削除については、「*既存のプールメンバの変更*」（4-4 ページ）を参照してください。

プールを削除する前に、まずバーチャルサーバからのリソースとしてのプールを削除する必要があります。

プールを削除するには

1. ナビゲーションペインの [Main] タブで、**[Local Traffic]** を展開して **[Pools]** をクリックします。
   [Pools] 画面が開きます。

2. プール名の左側にあるカラムで、[Select] ボックスをチェックします。
3. [**Delete**] をクリックします。
[Delete Confirmation] 画面が表示されます。
4. [**Delete**] をクリックします。

プールメンバの管理

BIG-IP では、既存のプールメンバを維持するために、プールメンバに特化したタスクを実行できるよう になっています。管理権限のあるプールメンバに対して、ユーザはプールメンバ一覧の確認、プールメンバプロパティの表示、プールメンバステータスの確認、プールメンバの有効化 / 無効化、プールメンバの削除を行えるよう になっています。

◆注

ユーザロールとパーティションアクセス割り当てに基づき、管理権限のあるプールメンバのみを管理することができます。

プールメンバのリストの表示

表示権限のあるプールメンバの一覧を表示することができます。プールメンバのリストを表示する場合、設定ユーティリティによって、各メンバに関する次の情報が表示されます。

- 状態
- IP アドレスとサービス
- Ratio
- プライオリティグループ

プールに定義されたプールメンバのリストを表示するには、次の手順に従います。

プールメンバのリスト を表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Pools**] をクリックします。
[Pools] 画面が開きます。
2. [Name] カラムで、該当するプールの名前をクリックします。
3. メニューバーで [**Members**] をクリックします。
プールのメンバがリストされます。

プールメンバプロパティの表示

各プールメンバの汎用プロパティを表示するには、設定ユーティリティを使用します。これらのプロパティとその説明を以下に示します。

- **アドレス**
関連付けられたノードの IP アドレス。

- **Service port**
  該当するサービスのポート番号。
- **パーティション**
  プールメンバが配置されたパーティション このパーティションとプール自体のパーティションは常に一致しています。
- **Parent node**
  プールメンバが関連付けられたノード（IP アドレス）。たとえば、プールメンバが **10.10.10.22:80** である場合、親ノードは **10.10.10.22** となります。
- **Availability**
  親ノード、プール、およびプールメンバが関連付けられたモニタに基づいた、プールメンバのステータス。
- **Health monitors**
  プールメンバに関連付けられたヘルスモニタ。
- **Current connections**
  プールメンバが受信した現在の接続数。
- **State**
  プールメンバで受信するトラフィックの状態。指定できる値は以下のとおりです。
  - **Enabled（すべてのトラフィックを許可）**
  - **Disabled（パーシステンスまたはアクティブ接続だけを許可）**
  - **Forced offline（アクティブ接続だけを許可）**

プールメンバプロパティを表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Pools**] をクリックします。
   [Pools] 画面が開きます。
2. [Name] カラムで、該当するプールの名前をクリックします。
3. メニューバーで [**Members**] をクリックします。
   プールのメンバがリストされます。
4. [Current Members] カラムで、プールメンバの IP アドレスとポート番号をクリックします。
   選択したプールメンバのプロパティが表示されます。

プールメンバステータスについて

プールメンバのステータスは、いつでも確認できます。設定ユーティリティを使用すると、プールメンバの **Availability** プロパティを表示して、この情報を確認できます。また、この情報は、プールメンバのリストを表示し、[Status] カラムを確認することによっても把握できます。

設定ユーティリティでは、複数のアイコンのいずれかを表示して、プールメンバのステータスを示します。これらのアイコンは、色や形で区別できます。ステータスアイコンの詳細については、表 4.4（ 4-25 ページ）を参照してください。設定ユーティリティ内でアイコンを表示するには、このページの「*プールメンバプロパティを表示するには*」

（ 4-24 ページ）を参照してください。ステータスアイコンの背景情報については、「*プールおよびプールメンバの管理*」（ 4-20 ページ）を参照してください。

◆ヒント

*プールメンバのアベイラビリティを手動で設定するには、関連するヘルスモニタの **Manual Resume** 属性を設定します。詳細については、第 12 章「**モニタの設定**」を参照してください。*

ステータス インジケータ	説明	State プロパティの設定
	プールメンバは **Enabled** に設定され、親ノードは **up** となり、モニタはプールメンバを **up** としてマークします。	**Enabled（すべてのトラフィックを許可）**
	プールメンバは使用できませんが、ユーザが処理を行わなくても、後で利用可能になる場合があります。このステータスは、同時接続数がプールメンバの **Connection Limit** 設定で定義された制限を超えた場合に発生します。	**Enabled（すべてのトラフィックを許可）**
	親ノードが down である、モニタがプールメンバを **down** としてマークした、またはユーザがプールメンバを向こうにしたため、プールメンバは利用できません。	**Enabled（すべてのトラフィックを許可）**
	プールメンバは [**Disabled**] に設定されていますが、モニタはプールメンバを **up** としてマークしています。通常の動作に復帰するには、手動でプールメンバを有効にする必要があります。	**Disabled（パーシステンスまたはアクティブ接続だけを許可）**
	プールメンバは [**Disabled**] に設定されており、親ノードが **down** であるため、オフラインになっています。通常の動作に復帰するには、手動でプールメンバを有効にする必要があります。	**Forced Offline（アクティブ接続だけを許可）**
	プールメンバは [**Disabled**] に設定されており、ユーザが無効にしたため、オフラインになっています。通常の動作に復帰するには、手動でプールメンバを有効にする必要があります。	**Disabled（パーシステンスまたはアクティブ接続だけを許可）**
	プールメンバは [**Disabled**] に設定されており、親ノードが **down** であるか、モニタがプールメンバを **down** としてマークしたため、オフラインになっています。通常の動作に復帰するには、手動でプールメンバを有効にする必要があります。	**Forced Offline（アクティブ接続だけを許可）**
	プールメンバまたはノードにモニタが関連付けられていないか、モニタの結果がまだ出力されていません。	**Enabled（すべてのトラフィックを許可）**

***表 4.4** プールメンバのステータスアイコンの説明*

◆注

*黒の状態アイコンは、関連モニタの **Manual Resume** 属性が有効になっていることを示しています。詳細については、第 12 章「**モニタの設定**」を参照してください。*

プールメンバの有効化または無効化

設定ユーティリティを使用すると、各プールメンバを有効または無効にできます。プールメンバを有効または無効にする場合、次のように、プールメンバの **State** プロパティの値が間接的に設定されます。

- **Enable**
  プールメンバの **State** プロパティを **Enabled（すべてのトラフィックを許可）**に設定します。
- **Disable**
  プールメンバの **State** プロパティを **Disabled（パーシステンスおよびアクティブ接続のみを許可）**に設定します。

無効になっているプールメンバと、モニタで **down** として報告されるプールメンバの違いは、無効になっているプールメンバがパーシステンスおよびアクティブ接続の処理を継続できることにあります。これとは異なり、**down** として報告されたプールメンバは、まったく接続を処理しません。

プールメンバリスト画面とプロパティ画面のステータスアイコンは、プールメンバが現在有効なっているか、無効になっているかを示します。プールメンバステータスの詳細については、「*プールメンバステータスについて*」（4-24 ページ）を参照してください。

プールメンバを有効または無効にするには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Pools**] をクリックします。
   表示権限を持つプールの一覧が表示されます。
2. 有効または無効にするプールメンバの IP アドレスとポート番号を見つけます。
3. 左側のカラムで [Select] ボックスをチェックします。
4. 画面の下部で、[**Enable**] または [**Disable**] をクリックします。

プールメンバの削除

既存のプールを削除するには、次の手順に従います。プールからの個々のプールメンバの削除については、「*既存のプールメンバの変更*」（4-4 ページ）を参照してください。

プールを削除する前に、まずバーチャルサーバからのリソースとしてのプールを削除する必要があります。

プールメンバを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Pools**] をクリックします。
   [Pools] 画面が開きます。
2. [Name] カラムで、プール名をクリックします。
   プールのプロパティが表示されます。
3. メニューバーで [**Members**] をクリックします。
   プールメンバのリストが表示されます。

4. プールメンバアドレスの左側にあるカラムで、[Select] ボックスをチェックします。
5. [**Delete**] をクリックします。
[Delete Confirmation] 画面が表示されます。
6. [**Delete**] をクリックします。

モニタの関連付けの削除

プールまたはプールメンバに対する既存のモニタの関連付けを削除できます。

プールからモニタを削除するには、プールのプロパティページにアクセスし、[**Active**] ボックス内のモニタ名を [**Available**] ボックス内に移動することで、[**Health Monitors**] 設定を変更します。

個々のプールメンバで明示的なモニタの関連付けを削除するには、プールメンバのプロパティページにアクセスし、[**Health Monitors**] 設定を[**Inherit from Pool**]または[**None**]のいずれかに変更します。[**None**] を選択すると、そのプールで設定しているすべての監視からプールメンバが除外されます。

プールおよびプールメンバの統計情報の表示

設定ユーティリティを使用して、既存のプールおよびプールメンバに関連した統計情報を表示できます。

プールおよびプールメンバの統計情報を表示するには、既存のプールのリストまたは既存のプールメンバのリストを表示します。次に、メニューバーで [**Statistics**] をクリックします。[Statistics] 画面が開いて、既存のプールおよびプールメンバすべての統計情報が表示されます。

表示される統計情報のタイプは次のとおりです。

- ビット（着信および発信）
- パケット（着信および発信）
- 接続（現在、最大、合計）

5

プロファイルについて

- プロファイルの概要
- プロファイルの作成と変更
- プロファイルの表示と削除
- プロファイルの実装
- 詳細情報

プロファイルの概要

BIG-IP® ローカルトラフィック管理システムでは、アプリケーション固有のネットワークトラフィックを、使用されるプロトコルやサービスに応じて、さまざまな方法で管理できます。たとえば、HTTP 応答データを圧縮したり、ターゲットサーバに要求を渡す前に SSL クライアント証明書を認証するように BIG-IP を設定できます。

BIG-IP には、管理対象のトラフィックのタイプ別に、そのトラフィックの動作をインテリジェントに制御するための設定ツールが用意されています。これらのツールはプロファイルと呼ばれます。*プロファイル*は、アプリケーション固有のトラフィックを管理するための機能を強化する、BIG-IP 提供の設定ツールです。より具体的に言うと、プロファイルとは、特定タイプのネットワークトラフィック（HTTP 接続など）の動作を制御するための、（デフォルト値を持つ）ユーザが構成可能な設定を含むオブジェクトのことです。プロファイルを設定したら、そのプロファイルをバーチャルサーバに関連付けます。この後、バーチャルサーバは、プロファイルに指定された値に基づいて、トラフィックを処理します。プロファイルを使用すると、ユーザはネットワークトラフィック管理の制御を強化したり、トラフィック管理作業をより容易かつ効率的に行うことが可能になります。

複数のプロファイルを 1 つのバーチャルサーバに関連付けることができます。たとえば、TCP プロファイル、SSL プロファイルおよび HTTP プロファイルを、同じバーチャルサーバに関連付けることができます。

プロファイルタイプ

BIG-IP では、複数のタイプのプロファイルが用意されています。プロファイルタイプには、特定のプロトコル（HTTP、SSL、FTP など）に対応するものと、複数のプロトコルに適用可能なトラフィック動作に関連するものがあります。例としては、接続パーシステンスプロファイル、認証プロファイルがあります。表 5.1 に、使用可能なプロファイルタイプを列挙して説明します。

プロファイルタイプ	説明
サービスプロファイル	
HTTP	HTTP トラフィックの動作を定義します。
FTP	FTP トラフィックの動作を定義します。
RTSP	RTSP トラフィックの動作を定義します。
SIP	Session Initiation Protocol（SIP）トラフィックの動作を定義します。 BIG-IP はこの SIP プロファイルに基づき SIP トラフィックの経路を決定し、SIP トラフィックは **Call-ID** でパーシストします。しかし、SIP パーシステンスプロファイルを作成することで、SIP トラフィックが **Call-ID** 以外の値でパーシストするように指定することも可能です（後述の*パーシステンスプロファイル*の項を参照）。
iSession	2 つの BIG-IP 間に、WAN により分離された最適化トンネルを作成します。

***表 5.1** BIG-IP で利用できるプロファイルのタイプ*

プロファイルタイプ	説明
パーシステンスプロファイル	
Cookie	HTTP Cookie を使用してセッションパーシステンスを実装します。
Destination Address Affinity	クライアントリクエストのヘッダで指定された宛先 IP アドレスに基づいて、セッションパーシステンスを実装します。スティッキーパーシステンスとも呼ばれます。
Hash	ユニバーサルパーシステンスと同様の方法でセッションパーシステンスを実装します。ただし、BIG-IP ではハッシュを使用してパーシステンスエントリを検索します。
Microsoft® Remote Desktop	Microsoft® Remote Desktop Protocol セッションのセッションパーシステンスを実装します。
SIP	SIP メッセージハンドリングを実装します。また、指定した SIP ヘッダフィールドに基づき、SIP パーシステンスの実装も行います。SIP パーシステンスプロファイルを使用するには、SIP プロファイルもあわせて作成する必要があります（前述の*サービスプロファイル*の項を参照）。
Source Address Affinity	クライアントリクエストのヘッダで指定された送信元 IP アドレスに基づいて、セッションパーシステンスを実装します。シンプルパーシステンスとも呼ばれます。
SSL	セッション ID を使用して、終了していない SSL セッションのセッションパーシステンスを実装します。
Universal	BIG-IP の Universal Inspection Engine（UIE）を使用してセッションパーシステンスを実装します。
プロトコルプロファイル	
Fast L4	レイヤ 4 の IP トラフィックの動作を定義します。
Fast HTTP	バーチャルサーバによるトラフィック処理速度を向上させます。
HTTP Class	ユーザ指定の基準を使用して、トラフィックのヘッダまたはコンテンツを調べることで、宛先にトラフィックを転送します。
TCP	TCP トラフィックの動作を定義します。
UDP	UDP トラフィックの動作を定義します。
SCTP	Streaming Control Transmission Protocol（SCTP）トラフィックの動作を定義します。
SSL プロファイル	
Client	クライアント側のSSLトラフィックの動作を定義します。「**パーシステンスプロファイル**」も参照してください。
Server	サーバ側の SSL トラフィックの動作を定義します。「**パーシステンスプロファイル**」も参照してください。
認証プロファイル	
LDAP	BIG-IP で、リモート Lightweight Directory Access Protocol（LDAP）サーバに保管されている認証データに基づいてトラフィックを認証できるようにします。
RADIUS	BIG-IP で、リモート RADIUS サーバに保管されている認証データに基づいてトラフィックを認証できるようにします。

***表 5.1** BIG-IP で利用できるプロファイルのタイプ（続き）*

プロファイルタイプ	説明
TACACS+	BIG-IP で、リモート TACACS+ サーバに保管されている認証データに基づいてトラフィックを認証できるようにします。
SSL Client Certificate LDAP	BIG-IP で、リモート LDAP サーバに保管されているデータに基づいて、サーバリソースへのクライアント のアクセスを制御できるよう にします。クライアント 承認資格情報は、SSL 証明書と 定義済みのユーザグループおよびロールに基づいています。
SSL OCSP	BIG-IP で、リモート Online Certificate Status Protocol（OCSP）サーバに保管されているデータを使用して、クライアント証明書の失効ステータスをチェックできるようにします。クライアントの資格情報は SSL 証明書に基づきます。
CRLDP	業界標準の Certificate Revocation List Distribution Point（CRLDP）技術を使い、証明書失効リスト（CRL）のステータスを管理します。
その他のプロファイル	
OneConnect	クライアントリクエストでサーバ側接続を再利用できるようにします。BIG-IP でサーバ側接続を再利用できる機能は、Connection Pooling™ と呼ばれています。
Statistics	ユーザ定義の統計情報カウンタを提供します。
NTLM	ユーザのパスワードをネットワークで送信することなく、暗号化された要求 / 応答プロトコルを使ってユーザの認証を行います。
Stream	TCP 接続などのデータストリーム内で文字列を検索および置換します。

***表 5.1** BIG-IP で利用できるプロファイルのタイプ（続き）*

デフォルトプロファイル

BIG-IP では、表 5.1 に列挙したプロファイルタイプごとに、1 つ以上のデフォルトプロファイルが用意されています。***デフォルトプロファイル***とは、その設定のデフォルト値を含む BIG-IP 提供のプロファイルのことです。デフォルトプロファイルの例として、**http** デフォルトプロファイルがあります。デフォルトプロファイルは、次のような方法で使用できます。

- **デフォルトプロファイルをそのまま使用する。**
  デフォルトプロファイルを参照するようにバーチャルサーバを設定できます。
- **デフォルトプロファイルの設定を変更する（非推奨）。**
  デフォルトプロファイルを変更すると、元のデフォルトプロファイルの設定が失われます。したがって、以降にそのデフォルトプロファイルに基づいて作成するカスタムプロファイルはすべて、変更された設定を継承することになります。
- **デフォルトプロファイルに基づいてカスタムプロファイルを作成する（推奨）。**
  この方法では、デフォルトプロファイルを保持し、カスタムプロファイル内でユーザ定義の設定を行えます。カスタムプロファイルは、指定した親プロファイルの設定値のいくつかを継承します。カスタムプロファイルの作成後、デフォルトプロファイルではなくカスタムプロファイルを参照するようにバーチャルサーバを設定できます。カスタムプロファイルの詳細については、次の「*カスタムプロファイルと親プロファイル*」を参照してください。

◆注

デフォルトプロファイルを変更することはできますが、作成したり削除することはできません。

カスタムプロファイルと親プロファイル

カスタムプロファイルは、ユーザが指定した親プロファイルから派生したプロファイルです。***親プロファイル***は、カスタムプロファイルが設定とそのデフォルト値を継承する元となるプロファイルです。

カスタムプロファイルを作成する場合、プロファイルが親プロファイルから継承した 1 つ以上の設定値を変更することができます。この場合、変更が必要な設定値と変更が必要がない設定値を選択できます。カスタムプロファイルを作成する利点には、親プロファイルの設定値をそのまま使用できることです。

◆注

カスタムプロファイルの作成時に親プロファイルを指定しない場合、BIG-IP では、対応するデフォルトプロファイルを親プロファイルとして自動的に割り当てます。たとえば、HTTP タイプのカスタムプロファイルを作成した場合、デフォルトの親プロファイルは ***http*** *となります。*

親プロファイルとしてのデフォルトプロファイルの使用

カスタムプロファイルの作成時に親プロファイルとして指定できる標準的なプロファイルは、デフォルトプロファイルです。たとえば、**my_http_profile** という HTTP タイプのカスタムプロファイルを作成した場合、デフォルトプロファイル **tcp** を親プロファイルとして使用できます。この場合、BIG-IP では、プロファイル **tcp** が自動的に作成され、このプロファイルには、デフォルトプロファイル **tcp** と同じ設定とデフォルト値が含まれます。このように、新しいカスタムプロファイルは、その親プロファイルから設定および値を継承します。この後、必要に応じて、カスタムプロファイルで継承した値をそのまま保持することも、変更することもできます。

親プロファイルとしてのカスタムプロファイルの使用

カスタムプロファイルを作成する場合、デフォルトプロファイルではなく別のカスタムプロファイルを親プロファイルとして指定できます。唯一の制約は、親として指定するカスタムプロファイルは、親から派生したプロファイルと同じタイプのプロファイルとする必要があるということです。新しいカスタムプロファイルを作成したら、その設定とデフォルト値は、親として指定したカスタムプロファイルから自動的に継承されます。

たとえば、**my_tcp_profile2** というプロファイルを作成する場合、カスタムプロファイル **my_tcp_profile** を親として指定できます。その結果、プロファイル **my_tcp_profile2** のデフォルト設定値は、その親プロファイル **my_tcp_profile** のものになります。

この後で親プロファイルの設定を変更した場合（**my_tcp_profile**）、BIG-IP ではこれらの変更が子プロファイルに自動的に伝播されます。

たとえば、カスタムプロファイル **my_tcpprofile** を作成し、これを親プロファイルとして使用してカスタムプロファイル**my_tcp_profile2**を作成する場合、親プロファイル **my_tcp_profile** に対して後で行った変更はすべて、プロファイル **my_tcp_profile2** に自動的に伝播されます。反対に、新しいカスタムプロファイル（この例では、**my_tcp_profile2**）の設定を変更した場合、新しいカスタムプロファイルは、これらの変更した設定の値を親プロファイルから継承しません。

プロファイルの要約

プロファイルは、特定タイプのネットワークトラフィックの動作に影響を与えるために使用できる設定ツールです。デフォルトでは、BIG-IP によって、そのまま使用できるプロファイルのセットが提供されます。これらのプロファイルには、異なるタイプのトラフィックの動作を定義するさまざまな設定が含まれています。プロファイルを使用して、接続およびセッションパーシステンスを有効にしたり、クライアントアプリケーション認証を管理することもできます。バーチャルサーバにプロファイルを割り当てている場合、BIG-IP により、そのプロファイルで定義されている設定に従って、プロファイルタイプに対応するトラフィックが管理されます。

プロファイルには、BIG-IP に付属するデフォルトプロファイルと、通常はユーザが作成するカスタムプロファイルの2つのタイプがあります。

特に HTTP および TCP トラフィックをより効率よく管理するため、BIG-IP には、F5 で作成した一連のカスタムプロファイルが用意されています。これらのプロファイルには、一般的によく使用する推奨設定が含まれています。これらのプロファイルを使用すれば、新しいプロファイルを作成する必要はありません。

デフォルトプロファイルは、それに含まれている値がユーザのニーズを満たしている場合に便利です。カスタムプロファイルは、デフォルトプロファイルに含まれている値とは異なる値をユーザが必要としている場合に便利です。プロファイルを設定および維持する作業を簡素化するために、BIG-IP では、カスタムプロファイルが親プロファイルの設定および値を自動的に継承するようになっています。

特定タイプのネットワークトラフィックを管理するためのプロファイルを作成した場合、これらのプロファイルは次のような方法で使用できます。

- デフォルトで有効になっているデフォルトプロファイルを使用するために処理を行う必要はありません。BIG-IP では、デフォルトプロファイルを使用して、これらのプロファイルで指定されている値に従って、対応するトラフィックタイプを自動的にダイレクトします。
- デフォルトプロファイルを親プロファイルとして使用して、カスタムプロファイルを作成し、そのプロファイルで定義されている値の一部またはすべてを変更できます。
- その他のカスタムプロファイルに対して親プロファイルとして使用するカスタムプロファイルを作成できます。

プロファイルの作成と変更

前の項で説明したように、プロファイルとは、アプリケーショントラフィックを管理するために役立つ設定ツールです。プロファイルを利用するには、BIG-IP で提供されるデフォルトプロファイルを使用するか、または独自のカスタムプロファイルを作成します。既存のプロファイルを必要に応じて変更することも可能です。

具体的には、次のことが可能です。

- デフォルトプロファイルをそのまま使用する。
- デフォルトプロファイルを変更する。
- カスタムプロファイルを作成する。
- カスタムプロファイルを変更する。

◆注

ユーザロールとパーティションアクセス割り当てに基づき、管理権限を持つプロファイルのみを管理することができます。

次の項では、プロファイルを作成および変更する手順を説明します。個々のプロファイル設定、およびそれらの設定がさまざまなタイプのトラフィックに与える効果については、この章の残りの部分または次の章のいずれかを参照してください。

- 第 6 章「*アプリケーション層トラフィックの管理*」
- 第 7 章「*セッションパーシステンスの有効化*」
- 第 8 章「*プロトコルプロファイルの管理*」
- 第 9 章「*SSL トラフィックの管理*」
- 第 10 章「*アプリケーショントラフィックの認証*」
- 第 11 章「*その他のプロファイルの使用*」

デフォルトプロファイルとカスタムプロファイルの背景情報については、「*プロファイルの概要*」（5-1 ページ）を参照してください。

デフォルトプロファイルをそのまま使用する

BIG-IP では、各タイプのトラフィックにそのまま使用できるデフォルトプロファイルが用意されています。デフォルトプロファイルには、そのタイプのトラフィックの管理に関連したプロパティおよび設定すべてのデフォルト値が含まれています。デフォルトプロファイルを実装するには、設定ユーティリティを使用して、プロファイルをバーチャルサーバに割り当てます。設定値を構成する必要はありません。詳細については、「*プロファイルの実装*」（5-12 ページ）を参照してください。

バーチャルサーバの作成または変更については、第 2 章「*バーチャルサーバの設定*」を参照してください。

デフォルトプロファイルを変更する

設定ユーティリティを使用して、デフォルトプロファイルの値を変更できます。この方法は推奨しません。デフォルトプロファイルの変更は、カスタムプロファイルの作成より単純ですばやく行えるように思えますが、これを行う場合、元の値が失われるという点に注意してください。プロファイルを元の状態にリセットするには、手動でデフォルトプロファイルの設定を再度変更して元の値を指定する必要があります （元のデフォルト値を確認するには、このガイド内の関連するプロファイルの章、またはオンラインヘルプを参照してください）。

◆注

*デフォルトプロファイルは、すべて **Common** パーティションに置かれます。詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

デフォルトプロファイルの変更および実装は、次の 2 つの作業から成るプロセスです。

- まず、設定ユーティリティを使用して、デフォルトプロファイルの設定を変更する必要があります。詳細については、次の項の「*デフォルトプロファイルを変更するには*」を参照してください。
- 次に、そのプロファイルをバーチャルサーバに関連付ける必要があります。バーチャルサーバへのプロファイルの関連付けについては、「*プロファイルの実装*」（5-12 ページ）を参照してください。

デフォルト プロファイルを変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [HTTP Profiles] 画面が開きます。
2. 変更するデフォルトプロファイルを選択します。
   - **http** プロファイルを変更する場合は、名前 **http** をクリックします。
     デフォルト **http** プロファイルのプロパティおよび設定が表示されます。
   - **http** プロファイル以外のデフォルトプロファイルを変更する場合は、メニューバーで該当するプロファイルメニューをクリックし、プロファイルタイプを選択します。ここでプロファイル名をクリックします。
     選択したデフォルトプロファイルのプロパティおよび設定が表示されます。
3. 必要に応じて設定を変更します。
4. [**Update**] をクリックします。

カスタムプロファイルを作成する

デフォルトプロファイルをそのまま、あるいは設定変更しても使用しない場合は、カスタムプロファイルを作成できます。カスタムプロファイルを作成してバーチャルサーバに関連付けることで、独自のトラフィック管理ポリシーのセットを実装できます。

カスタムプロファイルを作成する場合、そのプロファイルは子プロファイルであり、指定した親プロファイルの設定値を自動的に継承します。ただし、必要に応じて子プロファイル内の任意の値を変更できます。カスタムプロファイルおよび設定値の継承の背景情報については、「*カスタムプロファイルと親プロファイル*」（5-4 ページ）を参照してください。

親プロファイルを指定しない場合、BIG-IP では、作成するプロファイルのタイプと一致するデフォルトプロファイルを使用します。

◆ 重要

*カスタムプロファイルが作成されると、BIG-IP はそのプロファイルを現在の管理パーティションに配置します。パーティションの詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

カスタムプロファイルの実装は、次の 2 つの作業から成るプロセスです。

- まず、設定ユーティリティを使用して、カスタムプロファイルを作成する必要があります。詳細については、次の項の「*カスタムプロファイルを作成するには*」を参照してください。
- 次に、そのプロファイルをバーチャルサーバに関連付ける必要があります。バーチャルサーバへのプロファイルの関連付けについては、「*プロファイルの実装*」（5-12 ページ）を参照してください。

◆ 重要

設定ユーティリティ内で、各プロファイル作成画面には、各プロファイル設定の右側にチェックボックスがあります。設定のボックスをチェックしてからその設定の値を指定した場合は、以降に親プロファイル内の対応する値を変更しても、プロファイルではその値が保持されます。このように、設定のボックスをチェックしておくと、継承を通じて親プロファイルによってその値が上書きされることがありません。

カスタムプロファイルを作成するには

1. ナビゲーションペインの [Main] タブで [**Local Traffic**] を展開し、[**Profiles**] をクリックします。
   [Profiles] 画面が開いて、デフォルトで既存の HTTP プロファイルすべてのリストが表示されます。
2. 作成するプロファイルのタイプを選択します。
   - HTTP タイプのプロファイルを作成する場合は、ステップ 3 に進みます。
   - 別のタイプのプロファイルを作成する場合は、メニューバーのプロファイルカテゴリをクリックし、プロファイルタイプを選択します。

3. 画面の右側で [**Create**] をクリックします。
   新しいプロファイルを作成する画面が表示されます。

   ***注**: [Create] ボタンをクリックできない場合は、現在のユーザロールではプロファイルを作成する権限が与えられていないことを示します。*

4. [**Name**] ボックスで、プロファイルの一意の名前を入力します。
5. [**Parent Profile**] 設定で、リストからプロファイルを選択します。デフォルト プロファイルまたは別のカスタムプロファイルのいずれかを選択できます。
6. すべての設定の値を指定または変更するか、そのまま使用します。
   - 値を指定または変更する場合は、設定を見つけて、画面の右側にある [Custom] カラムでボックスをクリックし、値を入力または変更します。
   - 親プロファイルから継承された値をそのまま使用する場合は、設定をそのままにします。[Custom] カラムのボックスはチェックしないでください。
7. [**Finished**] をクリックします。

◆ヒント

*別の方法として、設定ユーティリティの [New Profile] 画面から [Main] タブにアクセスし、**[Local Traffic]** を展開し、**[Profiles]** メニュー項目の横にある **[Create]** ボタンをクリックし、プロファイルタイプを選択することもできます。*

カスタムプロファイルを変更する

カスタムプロファイルを作成している場合は、必要に応じて後で設定ユーティリティを使用して、カスタムプロファイルの設定を調整できます。プロファイルをバーチャルサーバにすでに関連付けている場合、その作業を繰り返す必要はありません。

◆重要

設定ユーティリティ内で、各プロファイル作成画面には、各プロファイル設定の右側にチェックボックスがあります。設定のボックスをチェックしてからその設定の値を指定した場合は、以降に親プロファイル内の対応する値を変更しても、プロファイルではその値が保持されます。このように、設定のボックスをチェックしておくと、継承を通じて親プロファイルによってその値が上書きされることがありません。

カスタムプロファイルの設定を変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [HTTP Profiles] 画面が開きます。
2. 変更するプロファイルタイプのメニュー（Services、Persistence、Protocols、SSL、または Authentication）から、プロファイルタイプを選択します。
   選択したタイプの既存のプロファイルのリストが表示されます。
3. [Name] カラムで、変更するプロファイルの名前をクリックします。
   選択したプロファイルの設定および値が表示されます。
4. すべての設定の値を変更するか、そのまま使用します。
   - 値を変更する場合は、設定を見つけて、画面の右側にある [Custom] カラムでボックスをクリックし、値を変更します。
   - 親プロファイルから継承された値をそのまま使用する場合は、設定をそのままにします。[Custom] カラムのボックスはチェックしないでください。
   - 値を親プロファイル値にリセットする場合は、画面の右側にある[Custom]カラムでチェックボックスをクリアします。
5. [**Update**] ボタンをクリックします。

プロファイルの表示と削除

設定ユーティリティを使用して、プロファイルのリストを表示したり、BIG-IP からプロファイルを削除することができます。

◆注

ユーザロールとパーティションアクセス割り当てに基づき、管理権限を持つプロファイルのみを管理することができます。

プロファイルのリストの表示

既存のプロファイルのリストを表示することができます。プロファイルのリストを表示すると、設定ユーティリティでは、各プロファイルに関する次の情報が表示されます。

- プロファイル名
- プロファイルが配置されたパーティション
- プロファイルのタイプ（パーシステンスおよび認証プロファイルのみ）
- 親プロファイル

BIG-IP で定義されたプロファイルのリストを表示するには、次の手順に従います。

◆ヒント

*既存のプロファイルをリスト表示する場合、プロファイルリストの上に表示される **[Search]** ボックスを使用できます。**[Search]** ボックスを使用すると、リストをフィルタリングする文字列を指定して、文字列と一致するオブジェクトのみを表示できます。デフォルト設定はアスタリスク（*****）で、すべてのオブジェクトが表示されます。*

プロファイルのリストを表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [HTTP Profiles] 画面が開きます。
2. HTTP プロファイル以外のプロファイルをリストする場合、ステップ 3 および 4 を実行します。
3. メニューバーで、表示するプロファイルのカテゴリをクリックします。たとえば、TCP プロファイルのリストを表示する場合、[**Protocol**] をクリックします。
4. メニューからプロファイルタイプを選択します。
   該当するプロファイルタイプのリスト画面が開きます。

プロファイルの削除

デフォルト プロファイルを除き、既存のプロファイルを削除できます。

プロファイルを削除するには

1. 前述の手順に従って、関連するプロファイルのリストを表示します。
2. 削除するカスタムプロファイルの左側にある [Select] ボックスをクリックします。
3. [**Delete**] をクリックします。
   確認画面が表示されます。
4. [**Delete**] をクリックします。

プロファイルの実装

特定タイプのトラフィックに対してプロファイルを作成している場合、そのプロファイルを 1 つ以上のバーチャルサーバに関連付けることでプロファイルを実装します。

プロファイルをバーチャルサーバに関連付けるには、そのプロファイルを参照するようにバーチャルサーバを設定します。バーチャルサーバがそのタイプのトラフィックを受信するたびに、BIG-IP によってそのトラフィックにプロファイル設定が適用されるので、その動作が制御されます。したがって、プロファイルによって、ネットワークトラフィックタイプごとに機能が定義されるだけでなく、これらの機能がバーチャルサーバでも利用可能になります。

プロファイルをバーチャルサーバに割り 当てるには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Virtual Servers**] をクリックします。
   [Virtual Servers] 画面が開きます。
2. [Name] カラムで、バーチャルサーバの名前をクリックします。
   選択したバーチャルサーバのプロパティおよび設定が表示されます。
3. 割り当てるプロファイルタイプの設定を見つけて、デフォルトプロファイルまたはカスタムプロファイルの名前を選択します。
4. 画面下部の [**Update**] をクリックします。

◆注

バーチャルサーバの作成時にバーチャルサーバにプロファイルを割り当てることもできます。

特定の種類のトラフィックでは複数のプロトコルおよびサービスを使用するため、ユーザは多くの場合、複数のプロファイルを作成して、単一のバーチャルサーバに関連付けます。

たとえば、クライアントアプリケーションで TCP、SSL、および HTTP プロトコルおよびサービスを使用して要求を送信する場合があります。そのため、このタイプのトラフィックでは、TCP、Client SSL、および HTTP に基づいた 3 つのプロファイルが必要になります。

各バーチャルサーバでは、そのバーチャルサーバに現在関連付けられているプロファイルの名前をリストします。ユーザは設定ユーティリティを使用して、プロファイルリストからプロファイルを追加または削除できます。BIG-IP では、特定のバーチャルサーバに対して許可されているプロファイルタイプの組み合わせに関して特定の要件があります。

トラフィックをダイレクトする際に、プロファイルリスト内に表示されない特定タイプのプロファイルをバーチャルサーバが必要とする場合は、BIG-IP によって関連するデフォルトプロファイルが使用され、プロファイルリストにプロファイルが自動的に追加されます。たとえば、クライアントアプリケーションから TCP、SSL、および HTTP を介してトラフィックが送信される場合、ユーザが SSL プロファイルおよび HTTP プロファイルのみを割り当てているときは、BIG-IP によってデフォルトプロファイル **tcp** がプロファイルリストに自動的に追加されます。

少なくとも、バーチャルサーバはプロファイルを参照する必要があり、そのプロファイルは UDP、FastL4、Fast HTTP、または TCP プロファイルタイプに関連付けられている必要があります。したがって、バーチャルサーバにプロファイルを関連付けていない場合、BIG-IP では、UDP、FastL4、Fast HTTP、または TCP デフォルトプロファイルがプロファイルリストに追加されます。

BIG-IP で選択されるデフォルトプロファイルは、バーチャルサーバのプロトコル設定によって異なります。プロトコルが **UDP** に設定されている場合、BIG-IP では、**udp** プロファイルがプロファイルリストに追加されます。プロトコルが **UDP** 以外に設定されている場合、プロファイルリストに **FastL4** プロファイルが追加されます。

詳細情報

各タイプのプロファイルの設定については、次の章を参照してください。

- 第 6 章「*アプリケーション層トラフィックの管理*」
- 第 7 章「*セッションパーシステンスの有効化*」
- 第 8 章「*プロトコルプロファイルの管理*」
- 第 9 章「*SSL トラフィックの管理*」
- 第 10 章「*アプリケーショントラフィックの認証*」
- 第 11 章「*その他のプロファイルの使用*」

6

アプリケーション層トラフィックの管理

- アプリケーション層トラフィック管理の概要
- HTTP 標準プロファイルの設定
- HTTP 圧縮の設定
- データ圧縮戦略の使い方
- RAM キャッシュ機能の設定
- データ圧縮と RAM キャッシュの最適化
- FTP プロファイルの設定
- SIP プロファイルの設定
- RTSP プロファイルの設定
- iSession プロファイルの設定

アプリケーション層トラフィック管理の概要

BIG-IP® ローカルトラフィック管理システムには、アプリケーション層トラフィックをインテリジェントに制御するためのいくつかの機能が備わっています。これらの機能の例としては、HTTP リクエストへのヘッダの挿入、HTTP サーバ応答の圧縮があります。

これらの機能は、各種設定プロファイルを通して利用できるようになっています。***プロファイル***とは、特定タイプのトラフィック（HTTP トラフィックなど）に対応する値を持つ設定のグループです。プロファイルでは、BIG-IP で対象のトラフィックタイプを管理する方法を定義します。

アプリケーション層トラフィックの管理は、以下のプロファイルタイプを使って行います。

- HTTP (Hyptertext Transfer Protocol)
- FTP (File Transfer Protocol)
- SIP (Session Initiation Protocol)
- RTSP（Real Time Streaming Protocol）
- iSession

◆注

*また、BIG-IP®Protocol Security Module のライセンスがある場合は、SMTP プロファイルを使用することができます。SMTP プロファイルの設定については、『**Configuration Guide for BIG-IP® Protocol Security Module**』を参照してください。*

これらのアプリケーション層プロファイルに加えて、BIG-IP にはアプリケーショントラフィックの管理に役立つその他の機能群が備わっています。これらの機能群には、HTTP サービスと FTP サービスの健全性をチェックするためのヘルスモニタや、ヘッダやコンテンツデータの問い合わせや操作を行うための iRules™ などがあります。詳細については、本ガイドの残りの章を参照してください。

◆注

*「Fast HTTP プロファイル」というプロファイルが存在し、これはHTTP プロファイルの一種ですが、Fast HTTP プロファイルの使用法は HTTP プロファイルのものとは異なります。Fast HTTP プロファイルの詳細については、第 8 章「**プロトコルプロファイルの管理**」を参照してください。*

HTTP 標準プロファイルの設定

HTTP プロファイルを設定すると、HTTP トラフィック管理を各自のニーズに合わせることができます。これらの設定は、設定ユーティリティの [New HTTP Profile] 画面上で、[General Properties]、[Settings]、[Compression]、[RAM Cache] のカテゴリに分けられています。プロファイルの作成時、またはプロファイルの作成後にプロファイルの設定を変更することで、これらの設定を行えます。プロファイルの設定に関する手順については、第 5 章「*プロファイルについて*」を参照してください。

[HTTP Profile] 画面の [Properties] および [Settings] 領域に表示されるプロファイル設定については、値が存在しない場合は指定するか、必要に応じてデフォルト値を変更することができます。その他の HTTP プロファイル設定については、「*HTTP 圧縮の設定*」（6-11 ページ）および「*RAM キャッシュ機能の設定*」（6-28 ページ）を参照してください。

BIG-IP では、以下の 4 種類の HTTP プロファイルを実装することができます。

- **http**
- **http-acceleration**
- **http-wan-optimized-compression**
- **http-lan-optimized-caching**
- **http-wan-optimized-compression-caching**

これらのプロファイルはそのまま実装することもできますし、ニーズに応じて設定値を変更して、カスタムプロファイルを作成することも可能です。

- 標準 HTTP プロファイルを実装する場合は、次の「*HTTP プロファイル設定について*」を参照してください。
- HTTP 圧縮機能を実装する場合は、「*HTTP 圧縮の設定*」（6-11 ページ）を参照してください。
- RAM キャッシュ機能を実装する場合は、「*RAM キャッシュ機能の設定*」（6-28 ページ）を参照してください。
- HTTP 最適化プロファイルを実装する場合は、「*データ圧縮と RAM キャッシュの最適化*」（6-34 ページ）を参照してください。

HTTP プロファイル設定について

デフォルトの **http** プロファイルをそのまま使用することも、カスタム HTTP プロファイルを作成することもできます。この **http** プロファイルは、親プロファイルから設定値を継承しないため、デフォルトプロファイルと見なされます。

表 6.1（6-3 ページ）に、HTTP タイプのプロファイルのプロファイル設定を示します。デフォルト値を持つ設定については、デフォルト設定をそのまま使用することも変更することもできます。この表の後に、設定の説明と、それらを変更するための手順を示します。

設定	説明	デフォルト値
Name	プロファイルのユーザ定義名を指定します。プロファイルの名前はユーザが指定する必要があります。	デフォルト値なし
Parent Profile	カスタムプロファイルの派生元とするプロファイルを指定します。	**http**
Basic Auth Realm	クライアント認証の認証レルムを指定します。	デフォルト値なし
Fallback Host	すべてのノードが停止した場合、もしくは選択したノードが停止して関連モニタがまだそのことを検出していない場合に、フォールバックホストとして HTTP **302** 応答を送信するように指定します。	デフォルト値なし
Fallback on Error Codes	[**Fallback Host**] 設定で指定したホストに HTTP 応答がリダイレクトされる原因となる HTTP エラーコードを指定します。	デフォルト値なし
Request Headers Insert	HTTP リクエストに挿入するヘッダ文字列を指定します。	デフォルト値なし
Request Headers Erase	HTTPリクエストから消去するヘッダ文字列を指定します。	デフォルト値なし
Response Headers Allowed	BIG-IP が HTTP 応答で許可するヘッダを指定します。	デフォルト値なし
Response Chunking	HTTP 応答に対するチャンクの処理方法を指定します。指定できる値は、[**Unchunk**]、[**Rechunk**]、[**Selective**]、および [**Preserve**] です。	**Selective**
OneConnect Transformations	接続を開いたままにするためにHTTPヘッダ変換を実行します。この機能では、One Connect™ プロファイルの設定が必要になります。	**Enabled**（チェックあり）
Redirect Rewrite	HTTPリダイレクションを変更できます。指定できる値は、[**Matching**]、[**All**]、[**Nodes**]、または [**None**] です。	**なし**
Encrypt Cookies	BIG-IP がクライアントシステムに送信する特定の Cookie を暗号化します。	デフォルト値なし
Cookie Encryption Passphrase	Cookie 暗号化のパスフレーズを指定します。	デフォルト値なし
Confirm Cookie Encryption Passphrase	[**Cookie Encryption Passphrase**] ボックスで指定したパスフレーズを確認します。	デフォルト値なし
Maximum Header Size	BIG-IP で HTTP ヘッダに許可する最大サイズをバイト単位で指定します。	**32768**
Pipelining	HTTP パイプライニングを有効または無効にします。	**Enabled**
Insert **XForwarded-For**	コネクションプーリングで使用するため、HTTP リクエストに **XForwarded-For** ヘッダを挿入します。この機能により、クライアントの IP アドレスが **XForwarded-For** ヘッダの値として追加されます。	**Disabled**
LWS Maximum Columns	HTTP リクエストに挿入される HTTP ヘッダに対して許可される最大幅を指定します。	**80**
LWS Separator	ヘッダが許可された最大幅を超える場合に、BIG-IPによって HTTP ヘッダ間で使用される区切り文字を指定します。	**\r\n**
Maximum Requests	単一の **Keep-Alive** 接続に許可される HTTP リクエストの最大数を指定します。	**0**

***表 6.1** HTTP プロファイルの設定*

設定	説明	デフォルト値
Protocol Security	チェックされている（有効になっている）場合、Protocol Security Module のセキュリティプロファイルを使って HTTPトラフィックのセキュリティ脆弱性をチェックします。このオプションは Protocol Security Module のライセンスがある場合にのみ利用可能です。	チェックしない

***表 6.1** HTTP プロファイルの設定（続き）*

HTTP プロファイルを設定する前に、変更する可能性がある特定の設定に関する説明を読んでおくと役立ちます。

プロファイル名の指定

HTTP プロファイルを作成するには、プロファイルの一意の名前を指定する必要があります。[**Name**] 設定は、HTTP プロファイルの作成時に値をアクティブに指定する必要がある 2 つの設定のうちの 1 つです。その他の設定にはすべて、デフォルト値があります。

プロファイル名を指定するには、[**Name**] 設定で、プロファイルの一意の名前を入力します。

親プロファイルの指定

作成する各プロファイルは、親プロファイルから派生します。デフォルトの **http** プロファイルを親プロファイルとして使用するか、すでに作成している別の HTTP プロファイルを使用することができます。

親プロファイルを指定するには、[**Parent Profile**] 設定でプロファイル名を選択します。

ベーシック認証のレルムの指定

[**Basic Auth Realm**] 設定の値は、ユーザが指定した文字列です。この文字列は、BIG-IP によってクライアント認証の一部としてクライアントに送信されます。

この設定を行うには、[**Basic Auth Realm**] 設定で値を入力します。

フォールバックホストの指定

HTTP プロファイル内で設定可能なもう 1 つの機能は、***HTTP リダイレクション***です。***HTTP リダイレクション***では、HTTP トラフィックを別のプロトコル識別子、ホスト名、ポート番号、または URI パスにリダイレクトできます。

フォールバックホストへのリダイレクションは、対象となるプールのすべてのメンバが利用できない場合、または選択したプールメンバが使用できない場合に発生します（***使用できない（利用不可）***という用語は、メンバが無効になっているか、**down** とマークされているか、対応するコネクションリミットを超えたことを示します）。1 つ以上のプールメンバが利用不可である場合、BIG-IP は、HTTP 応答 **Status Code 302 Found** により、HTTP リクエストをフォールバックホストへリダイレクトすることができます。

HTTP リダイレクションはしばしば、**LB_FAILED** iRule イベントが生成された場合にも発生しますが、以下のように、このイベントがなくてもリダイレクションが発生することがあります。

- **TCP3WHS** が完了した後、ノードが少なくともフル応答ヘッダを送信する前に選択したノードが **RST** を送信したとき
- BIG-IP が要求またはパイプライン要求の本文部分を受信している際に、選択したノードに到達できないと判断したとき

HTTP トラフィックがフォールバックホストにリダイレクトされるように BIG-IP を設定する場合、IP アドレスまたは完全修飾ドメイン名（FQDN）を指定できます。指定した値は、サーバによって応答で送信される **Location** ヘッダの値になります。たとえば、リダイレクションを **http://redirector.siterequest.com** として指定できます。

フォールバックエラーコードの指定

ターゲットサーバが利用不可能になった場合にトラフィックをリダイレクトするのに加え、フォールバックホストへのリダイレクトをトリガするサーバ応答の HTTP エラーコードを指定することができます。[**Fallback Error Codes**] 設定で指定する一般的なエラーコードには、**500**、**501**、**502** があります。

[**Fallback Error Codes**] ボックスにエラーコードを入力する際、コードとコードの間にスペースを入れるようにします（例 : **500 501 502**）。またエラーコードの範囲を指定することもできます（例 : **505-515**）。

HTTP リクエストへのヘッダの挿入

HTTP プロファイルにおけるオプションの設定は、HTTP ヘッダ挿入です。挿入する HTTP ヘッダには、クライアント IP アドレスを含めることができます。接続がセキュアネットワークアドレス変換（SNAT）を経由し、元のクライアント IP アドレスを保持する必要がある場合には、HTTP ヘッダにクライアント IP アドレスを含めると便利です。

一般に、ヘッダ挿入に指定するフォーマットは、引用符付き文字列です。ただし、Tool Command Language（Tcl）表現をヘッダに挿入して、希望の値にダイナミックに解決することもできます。設定済みの HTTP プロファイルをバーチャルサーバに割り当てた場合、プロファイルで指定されたヘッダは、BIG-IP によってプールまたはプールメンバに送信される任意の HTTP リクエストに挿入されます。

◆注

*クライアント IP アドレスなどの文字列を HTTP リクエストに挿入することに加えて、SSL 関連ヘッダを HTTP リクエストに挿入するように BIG-IP を設定できます。たとえば、クライアント証明書、暗号仕様、クライアントセッション ID などの挿入が可能です。これらのタイプのヘッダを挿入するには、iRule を作成する必要があります。iRule コマンドを使用してヘッダ挿入を実行する詳細については、第 17 章「**iRule の記述**」と、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。*

HTTP リクエストにヘッダを挿入するには、[**Request Headers Insert**] 設定で値を入力します。

HTTP ヘッダからのコンテンツの消去

もう 1 つのオプションの設定は、[**Header Erase**] 設定です。この設定を使用すると、クライアントからサーバに送信中の HTTP リクエストからヘッダのコンテンツを消去するようにプロファイルを設定できます。この機能では、ネットワークを介して HTTP リクエストを転送する前に、要求からヘッダコンテンツを消去できます。このようなヘッダには、情報が転送される前に消去する必要がある機密情報（ユーザ ID、電話番号など）が含まれていることがあります。

この設定を使用すると、指定されたヘッダのコンテンツが BIG-IP によって消去され、そのコンテンツが空白で置き換えられます。ヘッダ自体は保持されます。

◆注

この機能は、サーバからクライアントに送信中の HTTP 応答には適用されません。

消去対象のコンテンツを持つクライアントヘッダは、引用符付き文字列として指定する必要があります。HTTP リクエストからヘッダを消去するには、[**Request Headers Erase**] 設定で値を入力します。

HTTP レスポンスのヘッダの許可

設定ユーティリティを使って、BIG-IP で許可される HTTP レスポンスのヘッダを指定することができます。ヘッダを指定するには、[**Response Headers Allowed**] ボックスに値を入力します。複数のヘッダを指定する場合は、ヘッダとヘッダの間にスペースを入れます。たとえば、「**Content-Type Set-Cookie Location**」と入力すると、**Content-Type**、**Set-Cookie**、**Location** の 3 つのヘッダが許可されます。

チャンクの設定

HTTP 応答のコンテンツを圧縮する場合などでは、HTTP アプリケーションデータを検査したり変更したりする必要があります。このような検査や変更では、応答を***チャンク解除***する（つまり、チャンクエンコードしない）必要があります。[**Response Chunking**] 設定を使用すると、BIG-IP では、チャンクされた応答に対して処理を実行する前に、その応答をチャンク解除できます。

この設定の指定可能な値は、[**Unchunk**]、[**Rechunk**]、[**Selective**]、および [**Preserve**] です。デフォルト値は [**Selective**] です。

表 6.2 に、これらの値のそれぞれを示し、元の応答がチャンクされているかチャンク解除されているかに応じて、BIG-IP で行われる処理を説明します。

設定	元の応答がチャンクされている場合	元の応答がチャンク解除されている場合
Unchunk	BIG-IP では応答をチャンク解除し、HTTP コンテンツを処理し、応答をチャンク解除されているものとして渡します。すべてのデータがクライアントに送信されると（**Connection: Close** ヘッダで示されます）接続が終了します。	BIG-IP では HTTP コンテンツを処理し、応答をそのまま渡します。
Rechunk	BIG-IP では応答をチャンク解除し、HTTP コンテンツを処理し、チャンクトレーラヘッダを再追加した後、応答をチャンクされているものとして渡します。チャンク拡張はすべて失われます。	BIG-IP では、転送エンコードおよびチャンクヘッダを出力側で追加します。
Selective	[**Rechunk**] と同じです。	BIG-IP では HTTP コンテンツを処理した後、応答をそのまま渡します。
Preserve	BIG-IP では応答をチャンクされたままにし、HTTP コンテンツを処理し、応答をそのまま渡します。HTTP 圧縮が有効になっている場合でも、BIG-IP では応答を圧縮しません。	BIG-IP では HTTP コンテンツを処理した後、応答をそのまま渡します。

***表 6.2** BIG-IP のチャンク動作*

OneConnect 変換の有効化または無効化

この設定は OneConnect™ 機能の一部を有効または無効にするものであり、HTTP/1.0 接続に対してのみ適用されます。この設定が有効になっており、バーチャルサーバに OneConnect プロファイルが割り当てられている場合、クライアント接続を開いたままにしておくために、**Connection** ヘッダの変換が実行されます。具体的には以下のように変換されます。

1. クライアントが HTTP/1.0 要求を送信します。
2. サーバが、先頭に **Connection: Close** ヘッダがある応答を送信します。
3. BIG-IP が **Connection: Close** ヘッダを **Connection: Keep-Alive** に変換します。
4. OneConnect プロファイルの利用を通して、サーバ側の接続が切り離され、ほかの要求の対応に使われている有効なサーバ側の接続のプールに送られた後、最終的にクローズします。このプロセスはクライアントからは見えないようになっています。
5. クライアント側の接続は開いたままとなり、サーバ側の接続が開いていて、そのクライアントからのさらなる要求を受け入れることができるという前提のもと動作します。

この設定のデフォルト値は、[**Enabled**] です。OneConnect 変換の有効化 / 無効化を行うには、[**OneConnect Transformations**] 設定を有効にします。

◆ 重要

*この設定を有効にするには、OneConnect™ プロファイルをあわせて設定し、コネクションプーリングを有効にする必要があります。コネクションプーリングおよび OneConnect プロファイルの設定の詳細については、第 5 章「**プロファイルについて**」を参照してください。*

OneConnect™ 機能の概要については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。

HTTP リダイレクションの書き換え

場合によっては、クライアントリクエストは HTTPS プロトコルから、ノンセキュアなチャンネルである HTTP プロトコルにリダイレクトされることがあります。要求をセキュアなチャンネルで維持するには、元の HTTPS プロトコルにリダイレクトされるように、リダイレクションが書き換えられるようにします。

リダイレクションの書き換えは、リダイレクション応答の HTTP **Location** ヘッダのみで行われ、リダイレクションのコンテンツでは行われません。

BIG-IP で HTTP リダイレクションを書き換えられるようにするには、設定ユーティリティを使用して、書き換え時の URI の処理方法を指定します。この機能が有効になっている場合、プロトコル名とポート番号が書き換えられます。

この設定の指定可能な値は、[**Matching**]、[**All**]、[**Nodes**]、または [**None**] です。

書き換える URI の選択

HTTP リダイレクションを書き換えるように BIG-IP を設定する場合、最初にクライアントによって要求された URI と一致する URI のみ（オプションの末尾のスラッシュを省いたもの）を書き換えるか、すべての URI を書き換えるかを指定します。後者の場合は、常にリダイレクト先の URI が書き換えられます。これらの URI は、最初に要求された URI と一致する場合と同様に書き換えられます。

URI にホスト名ではなくノード IP アドレスが含まれる場合は、その IP アドレスをバーチャルサーバアドレスに変更するように BIG-IP を設定できます。

表 6.3 に、BIG-IP がポート **443** をリスンし、[**Rewrite Redirections**] 設定が有効になっている場合に、クライアントリクエストのリダイレクションがどのように変換されるかの例を示します。

元のリダイレクション	リダイレクションの書き換え
http://www.myweb.com/myapp/	**https://www.myweb.com/myapp/**
http://www.myweb.com:8080/myapp/	**https://www.myweb.com/myapp/**

***表 6.3** BIG-IP がポート **443** をリスンしている場合の HTTP リダイレクションの書き換え例*

表 6.4 に、BIG-IP がポート **4443** をリスンし、書き換え機能が有効になっている場合、クライアントリクエストのリダイレクションがどのように変換されるかの例を示します。

元のリダイレクション	リダイレクションの書き換え
http://www.myweb.com/myapp/	**https://www.myweb.com:4443/myapp/**
http://www.myweb.com:8080/myapp/	**https://www.myweb.com:4443/myapp/**

***表 6.4** BIG-IP がポート **4443** をリスンしている場合の HTTP リダイレクションの書き換え例*

プロトコル名の書き換え

HTTP リダイレクションを書き換えるように設定されている場合、BIG-IP によって HTTP プロトコル名が HTTPS に書き換えられます。たとえば、クライアントが要求を **https://www.sample.com/bar** に送信し、ノンセキュアなチャンネルである **http://www.sample.com/bar/** に最初にリダイレクトされる場合があります。クライアントリクエストをセキュアなチャンネルに維持する場合は、リダイレクトされた URI を書き換えて **https://www.sample.com/bar/** に移動するように、BIG-IP を設定できます （末尾のスラッシュが追加されていることに注意してください）。

Cookie の暗号化と復号化

設定ユーティリティを使って、BIG-IP がクライアントシステムに送信する複数の Cookie を暗号化することができます。暗号化された Cookie がクライアントから返信されると、BIG-IP はそれらの Cookie を復号化します。

Cookie を暗号化するには、[**Encrypt Cookie**] 設定を使って、BIG-IP で暗号化する Cookie の名前を指定します。複数の Cookie を指定する場合は、Cookie と Cookie の間にスペースを入れます。

Cookie 名の指定が終わったら、[**Cookie Encryption Passphrase**] ボックスと [**Confirm Cookie Encryption Passphrase**] ボックスを使って、その Cookie のパスフレーズを入力します。

最大ヘッダサイズの指定

Maximum Header Size 設定では、BIG-IP で HTTP ヘッダに許可される最大サイズを指定できます。デフォルト値は **32768**（バイト単位）です。

パイプライニングのサポートの有効化

通常は、前の要求が応答を受信するまで、クライアントが要求を開始することはできません。HTTP/1.1 *パイプライニング*を使用した場合、前の要求が応答を受信していなくても、クライアントは複数の要求を開始することができます。

ただし、開始された要求が連続して処理されることに変わりはなく、前の要求が応答を受信するまでキュー内の要求が処理されることはありません。

BIG-IP でパイプライニングのサポートを有効にしておけば、宛先サーバでパイプライニングを有効にする必要はありません。

パイプライニングを有効化 / 無効化するには、**[Pipelining]** 設定でボックスをチェックします。デフォルトでは、この機能は **[Enabled]** に設定されています。

XForwarded For ヘッダの挿入

コネクションプーリングを使用してクライアントが既存のサーバ側接続を利用できるようにする場合、**XForwarded For** ヘッダを要求に挿入できます。このヘッダを挿入するように BIG-IP を設定すると、ターゲットサーバでは、接続を開始したクライアント以外のクライアントから来る要求を識別できます。デフォルトでは **[Disabled]** に設定されています。

LWS 最大カラムの設定

[LWS Maximum Columns] 設定では、HTTP リクエストに挿入されるヘッダに対して許可される最大カラム数を指定します。

[LWS Maximum Columns] を設定するには、最大値を指定します。この設定のデフォルト値は、**[80]** です。

LWS セパレータの設定

[LWS Separator] 設定では、ヘッダが **[LWS Maximum Columns]** 設定で指定されている最大幅を超える場合に、BIG-IP が HTTP ヘッダ間で使用する区切り文字を指定します。

[LWS Separator] 設定を設定するには、区切り文字の値を指定します。この設定にはデフォルト値はありません。

最大リクエスト数の指定

[Maximum Requests] 設定では、単一の Keep-Alive 接続に許可する要求の最大数を指定します。指定された制限に達した場合、最終応答には **Connection: close** ヘッダが含まれ、その後に接続のクローズが続きます。デフォルト設定の **0** を指定した場合、**Keep-Alive** 接続ごとに無限数の要求が許可されます。

HTTP 圧縮の設定

標準的なクライアントとサーバのシナリオでは、HTTP コンテンツを圧縮および圧縮解除するようにブラウザとサーバを設定できます。HTTP 圧縮によって送信データの量が少なくなるので、帯域幅の使用量が大幅に抑えられます。次の 2 つの項では、HTTP 圧縮タスクを実行するように BIG-IP を設定する場合の利点について説明します。

標準的なクライアントとサーバのシナリオにおける圧縮

クライアントとサーバの環境で HTTP 圧縮が有効になっている場合、ブラウザは**Accept-Encoding**ヘッダをクライアントリクエストに挿入し、ブラウザで認識可能な圧縮方式を指定します。その結果、サーバはヘッダを読み取り、圧縮方式の 1 つを使用して応答本文を圧縮します。その後サーバは、**Content-Encoding** ヘッダを応答に挿入して、サーバによって使用された圧縮方式をブラウザに通知します。ブラウザは、圧縮された応答を受信すると、**Content-Encoding** ヘッダを読み取り、それに従ってデータを圧縮解除します。

BIG-IP を使用せずに HTTP 圧縮を有効にする場合は通常、宛先サーバ上に圧縮ソフトウェアをインストールして設定する必要があります。

BIG-IP を使用した圧縮

BIG-IP のオプション機能として、ターゲットサーバから HTTP 圧縮タスクの負荷を軽減できます。BIG-IP 上で HTTP 圧縮を設定するために必要なすべての作業および圧縮ソフトウェア自体は、BIG-IP に集約されています。

データ圧縮の設定時には、以下のこともあわせて行えます。

- 特定のタイプのデータを包含するか除外するかを設定
- 希望する圧縮の質と速度のレベルを指定

HTTP 圧縮オプションを有効にする主な方法は、HTTP プロファイルの [**Compression**] 設定を [**Enabled**] に設定することです。この設定を行うと、HTTP プロファイルの [**URI List**] 設定もしくは [**Content List**] 設定で指定した値が [**Request-URI**] 応答ヘッダか [**Content-Type**] 応答ヘッダの値と一致する、あらゆる応答の HTTP コンテンツが圧縮されることになります。

◆注

*[**URI List**] 設定や [**Content List**] 設定では、パターン文字列もしくは正規表現で文字列を指定するようにします。リストタイプでは、パターン文字列の大文字と小文字が区別されます。たとえば、**www.f5.com** というパターン文字列は **www.F5.com** とは別のものとして扱われます。大文字と小文字の区別を無効にするには、Linux **regexp** コマンドを使用します。*

◆ 重要

*HTTP プロファイルを設定して圧縮を有効にした場合、「Speed」というデフォルトの圧縮戦略が使用されることになります。Speed 圧縮戦略では、BIG-IP は可能なかぎりハードウェア圧縮プロバイダを利用して、パフォーマンスを最大化します。圧縮戦略を変更するにはコマンドラインインターフェイスを使用します。使用可能な圧縮戦略の詳細については、「**データ圧縮戦略の使い方**」(6-22 ページ) を参照してください。*

特定の接続に対して HTTP 圧縮を有効にするには、**HTTP:compress enable** コマンドを指定する iRule を作成します。詳細については、第 17 章「*iRule の記述*」と、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。

HTTP 圧縮が BIG-IP で有効になっている場合、BIG-IP では、一連の手順が実行されます。

1. まず、BIG-IP はクライアントリクエストの **Accept-Encoding** ヘッダを読み取り、**deflate** または **gzip** 圧縮方式のいずれかの仕様を検索し、いずれかの方式が優先としてマークされているかどうかを調べます。
2. HTTPプロファイルの[**Keep Accept Encoding**]設定が[**Disabled**]に設定されている場合、BIG-IP は要求から **Accept-Encoding** ヘッダを削除し、その要求をサーバに渡します。
Accept-Encoding ヘッダを削除すると、サーバによって HTTP 圧縮が実行されたり、**Content-Encoding** ヘッダが応答に挿入されることを回避できます。
3. BIG-IP は、サーバ応答を受信すると、選択している圧縮方式を指定する **Content-Encoding** ヘッダを挿入します。BIG-IP は、クライアントリクエストの **Accept-Encoding** ヘッダで **gzip** または **deflate** のいずれかの圧縮方式の仕様を検索することで、圧縮方式を選択します。クライアントリクエストで圧縮方式が指定されていない場合、BIG-IP は **gzip** または **deflate** 方式を使用して応答データを圧縮します。この場合、BIG-IP で使用されるデフォルトの方式は **gzip** となります。
4. 最後に、BIG-IP は応答を圧縮し、クライアントに送信します。その後クライアントは、応答の **Content-Encoding** ヘッダを読み取り、使用されている圧縮方式を判断し、それに従ってデータを圧縮解除します。

BIG-IP の HTTP 圧縮機能を使用する場合、指定した特定のタイプの URI またはファイルを含めることも、除外することもできます。URI やファイルタイプによってはすでに圧縮されているものもあるため、除外が行えれば便利です。すでに圧縮されているデータの圧縮に CPU リソースを使用することはお勧めしません。通常そのコストがメリットを上回るためです。除外に対して指定できる正規表現の例には、**.*\.pdf**、**.*\.gif**、**.*\.html** があります。

データ圧縮のその他の設定方法

BIG-IP にデータ圧縮を実装する場合、第一の方法はカスタム HTTP プロファイルを作成することですが、ほかにも以下のような方法でデータ圧縮を実装することができます。

- デフォルトの **http-wan-optimized-compression** プロファイルをそのまま実装することができます。このプロファイルには、データ圧縮を最適化するための設定が最初から含まれています。この場合、設定の手間は省けますが、圧縮対象のデータタイプを細かく制御することができません。詳細については、「*http-wan-optimized-compression プロファイルの使い方*」（6-35 ページ）を参照してください。
- **http-wan-optimized-compression** プロファイルをもとにカスタムプロファイルを作成し、圧縮設定を変更することができます。詳細については、「*http-wan-optimized-compression プロファイルの使い方*」（6-35 ページ）を参照してください。
- より高度な圧縮実装を行うには、デフォルトの **http** プロファイルをもとにカスタムプロファイルを作成した後に、コマンドラインインターフェイスを使って別の圧縮戦略を設定します。圧縮戦略が違えば、できるかぎり効率的な圧縮を提供するために、システムがハードウェアおよびソフトウェア圧縮プロバイダを使用する方法も変わってきます。詳細については、「*データ圧縮戦略の使い方*」（6-22 ページ）を参照してください。

データ圧縮への HTTP プロファイルの使用

前述したように、データ圧縮を実装するひとつの方法が、デフォルトの **http** プロファイルに基づいてカスタムプロファイルを作成することです。表 6.5 に、作成したカスタム HTTP プロファイル内で指定可能な圧縮設定を示します。これらの設定を行う場合は、デフォルト値が存在しない場合に値を指定するか、またはデフォルト値を変更します。

設定	説明	デフォルト値
Compression	HTTP 圧縮機能を有効または無効にします。	**Disabled**
URI Compression	特定の **Request-URI** 要求を包含または除外するための設定を表示します。指定できる値は、[**URI List**] または [**Not Configured**] です。	**Not Configured**
URI List	[**URI Compression**] 設定が [**Enabled**] に設定されている場合、圧縮対象の URI、および URI 圧縮に包含したり URI 圧縮から除外する応答のタイプを指定します。 **注 :** *パターン文字列を使用する場合、リストタイプの大文字と小文字が区別されます 詳細については、本章の前の項を参照してください。*	デフォルト値なし
Content Compression	特定の **Content-Type** 応答を包含または除外するための設定を表示します。指定できる値は、[**Content List**] または [**Not Configured**] です。	**Not Configured**

***表 6.5** HTTP プロファイルの圧縮設定*

設定	説明	デフォルト値
Content List	[**Content Compression**] 設定が [**Enabled**] に設定されている場合、圧縮対象のコンテンツのタイプ、およびコンテンツ圧縮に包含したりコンテンツ圧縮から除外する応答のタイプを指定します。 **注 :** *パターン文字列を使用する場合、リストタイプの大文字と小文字が区別されます 詳細については、本章の前の項を参照してください。*	[**Include List**] ボックスでは、デフォルト値は次のとおりです。 **text/ application/(xml|x-javascript)**
Preferred Method	応答を圧縮するために使用する圧縮方式を指定します。指定できる値は、[**gzip**] および [**deflate**] です。	**gzip**
Minimum Content Length	圧縮対象として受け付け可能なサーバ応答の最小長（バイト 単位）を指定します。この長さは、コンテンツの長さのみに適用され、ヘッダには適用されません。	**1024**
Compression Buffer Size	圧縮サイズを指定する **Content-Length** ヘッダを応答に挿入するかどうかが BIG-IP によって決定される前に、バッファに格納される最大圧縮バイト数を指定します。	**4096**
gzip Compression Level	圧縮の量とレートを指定します。	**1 - Least Compression (Fastest)**
gzip Memory Level	サーバ応答の圧縮時に、BIG-IP が内部圧縮バッファに使用するメモリをキロバイト単位で指定します。	**8**
gzip Window Size	サーバ応答の圧縮時に BIG-IP が使用するウィンドウサイズをキロバイト単位で指定します。	**16**
Vary Header	キャッシュ可能なサーバ応答への **Vary** ヘッダの挿入を有効または無効にします。	**Enabled**
HTTP/1.0 Requests	HTTP/1.0 クライアントリクエストへの応答の圧縮を有効または無効にします。	**Disabled**
Keep Accept Encoding	有効になっている場合、ターゲットサーバは BIG-IP の代わりに HTTP 圧縮を実行できます。	**Disabled**
Browser Workarounds	ブラウザの回避策を実装します。	Disabled（解除）
CPU Saver	チェックされている場合（有効になっている場合）、CPU 使用率が監視され、CPU 使用率が [**CPU Saver High Threshold**] または [**CPU Saver Low Threshold**] のいずれかに達すると、圧縮速度が自動的に調整されます。	**Enabled**（チェックあり）
CPU Saver High Threshold	圧縮対象のコンテンツ量および適用される圧縮量が変更されるようにする CPU 使用量を指定します。	**90**
CPU Saver Low Threshold	ユーザ定義の圧縮値が復元されるようにする CPU 使用量を指定します。	**75**

***表 6.5** HTTP プロファイルの圧縮設定（続き）*

HTTP プロファイルを設定する前に、変更する可能性がある圧縮設定に関する説明を読んでおくと役立ちます。

圧縮機能の有効化または無効化

[**Compression**] 設定を指定すると、BIG-IP によってサーバ応答の圧縮が実行されます。指定できる値は次のとおりです。

- **Enabled**
  この値を設定すると、HTTP プロファイルの [**URI List**] または [**Content List**]設定で指定した値と一致する応答すべてのHTTPサーバコンテンツの圧縮が BIG-IP によって実行または抑止されます。
- **Disabled**
  [**Compression**] 設定が [**Disabled**] に設定されている場合、BIG-IP では HTTP 圧縮が実行されません。
- **Selective**
  この値を設定すると、**HTTP::compress** コマンドを含むiRuleによって指定された場合のみ、BIG-IP で HTTP 圧縮が実行されます。

HTTP 圧縮を有効にするには、[**Compression**] 設定で [**Enabled**] を選択します。iRule コマンド **HTTP::compress <enable>** を使用して圧縮を有効にする場合は、[**Selective**] を選択します。

URI 圧縮の使用

圧縮を有効にする場合、BIG-IP によって特定の種類のサーバ応答が圧縮されないように指定できます。[**URI Compression**] 設定を使用して、その値を [**URI List**] に設定することで、クライアントリクエストの URI で指定されている特定の応答を圧縮に包含したり圧縮から除外するように BIG-IP に指示できます。

具体的には、正規表現を入力することで、BIG-IP で圧縮に包含したり圧縮から除外したりするサーバ応答のタイプを指定できます。たとえば、正規表現 **.*.htm** を入力することで、すべての **.htm** 応答を BIG-IP で圧縮するように指定できます。この場合、BIG-IP により、該当する応答タイプが各クライアントリクエスト内で指定されている URI と比較され、一致が検出されると、処理が行われます。

正規表現はすべて、Advanced Regular Expression（ARE）構文で指定する必要があります。

URI 圧縮機能を使用するには、[**URI Compression**] 設定で [**URI List**] を選択します。その後、[**Include List**] ボックスまたは [**Exclude List**] ボックスを使用して、正規表現を入力します。リスト（URI またはコンテンツ）を指定しない場合、BIG-IP ではすべての応答が圧縮されます。コンテンツリストの詳細については、「*コンテンツ圧縮の使用*」（6-16 ページ）を参照してください。

HTTP 圧縮への URI 指定応答の包含

[**URI compression**] *設定が有効になっている場合、*[**Include List**] *に 1 つ以上の値を入力すると、クライアントリクエスト行の URI 部分と一致する応答のみが BIG-IP によって圧縮されます。*

[**Include List**] ボックスでは、値を正規表現の形式で指定します。たとえば、[**Include List**] ボックスに値 **.*.txt**、**.*.htm**、および **.*.html** が含まれており、これらの表現がクライアントリクエスト内の URI と一致する場合、指定された正規表現と一致する URI に対する応答のみが BIG-IP によって圧縮されます。

この設定を問題なく適用するために、BIG-IP では [**Include List**] ボックスで指定されている値の少なくとも1つで一致を検出する必要があります。BIG-IP によって一致が検出されない場合、応答は圧縮されません。

HTTP 圧縮からの URI 指定応答の除外

[**URI compression**] 設定が有効になっている場合、[**Exclude List**] に 1 つ以上の値を入力すると、クライアントリクエスト行の URI 部分と一致する応答が BIG-IP によって圧縮から除外されます。

[**Exclude List**] ボックスでは、値を正規表現の形式で指定します。たとえば、[**Exclude List**] ボックスに値 **.*.pdf** が含まれており、その表現がクライアントリクエスト内の URI と一致する場合、これらの URI に対する **.pdf** 応答はすべて、BIG-IP によって圧縮から除外されます。

この設定を問題なく適用するために、BIG-IP では [**Exclude List**] ボックスで指定されている値の少なくとも1つで一致を検出する必要があります。BIG-IP によって一致が検出されない場合、応答は圧縮から除外されません。

コンテンツ圧縮の使用

圧縮を有効にする場合、BIG-IP によって特定の種類のサーバ応答が圧縮されないように指定できます。[**Content Compression**] 設定を使用すると、その値を [**Content List**] に設定して、サーバ応答の **Content-Type** ヘッダで指定されている特定の応答を圧縮に包含したり圧縮から除外するように、BIG-IP に指示できます。

具体的には、正規表現を入力することで、BIG-IP で圧縮に包含したり圧縮から除外したりするサーバ応答のタイプを指定できます。たとえば、正規表現 **.*.htm** を入力することで、すべての **.htm** 応答を BIG-IP で圧縮するように指定できます。この場合、BIG-IP により、該当する応答タイプが各サーバ応答内で指定されている **Content-Type** ヘッダの値と比較され、一致が検出されると、処理が行われます。

正規表現はすべて、Advanced Regular Expression（ARE）構文で指定する必要があります。

コンテンツ圧縮機能を使用するには、[**Content Compression**] 設定で [**Content List**] を選択します。その後、[**Include List**] ボックスまたは [**Exclude List**] ボックスを使用して、正規表現を入力します。リスト（URI またはコンテンツ）を指定しない場合、BIG-IP ではすべての応答が圧縮されます。コンテンツリストの詳細については、「*URI 圧縮の使用*」（6-15 ページ）を参照してください。

HTTP 圧縮への Content-Type 応答の包含

圧縮が有効になっている場合、[**Include List**] ボックスで 1 つ以上の値を指定すると、サーバの **Content-Type** ヘッダの値と一致する応答のみが BIG-IP によって包含されます。この設定の値は、これらのヘッダ値のリストです。たとえば、[**Include List**] ボックスに値 **application/pdf** および **image/**** が含まれている場合、これらのコンテンツタイプの応答のみが圧縮されます。

すべてのテキストタイプを包含するには、この設定に値 **text/.*** を割り当てます。

HTTP 圧縮からの Content-Type 応答の除外

圧縮が有効になっている場合、[**Exclude List**] ボックスで 1 つ以上の値を指定すると、サーバの **Content-Type** ヘッダの値と一致する応答が BIG-IP によって除外されます。この設定の値は、これらのヘッダ値のリストです。たとえば、[**Exclude List**] ボックスに値 **application/pdf** および **image/**** が含まれている場合、これらのコンテンツタイプの応答は圧縮から除外されます。

すべてのテキストタイプを除外するには、この設定に値 **text/.*** を割り当てます。

優先圧縮方式の指定

[**Preferred Method**] 設定を使用すると、応答の圧縮時に BIG-IP で使用する圧縮方式を指定できます。2 つの指定可能な値は、[**gzip**] および [**deflate**] です。デフォルト値は [**gzip**] です。

圧縮に対する最小コンテンツ長の指定

圧縮が有効になっている場合、[**Minimum Content Length**] 設定では、BIG-IP で圧縮対象のサーバ応答に必要とされる最小長（非圧縮バイト数）を指定します。BIG-IP により、サーバ応答の **Content-Length** ヘッダでサーバ応答のコンテンツ長が検出されます。したがって、応答ヘッダで指定されているコンテンツ長が [**Minimum Content Length**] 設定に割り当てられている値より小さい場合、BIG-IP は応答を圧縮しません。バイト単位の長さは、コンテンツの長さのみに適用され、ヘッダには適用されません。

たとえば、デフォルト値 **1024** を使用すると、少なくとも 1024 バイトを含む HTTP コンテンツを持つ応答のみが BIG-IP によって圧縮されます。

場合によっては、**Content-Length** ヘッダに応答のコンテンツ長が示されないことがあります。このような場合は、サイズにかかわらず、BIG-IP では応答が圧縮されます。

最小のコンテンツ長を指定するには、[**Minimum Content Length**] 設定で数値を入力します。

圧縮バッファサイズの指定

圧縮が有効になっている場合、[**Compression Buffer Size**] 設定では、**Keep-Alive** 接続を保持して **Content-Length** ヘッダを書き換えるかどうかが BIG-IP によって決定される前に、バッファに格納される最大圧縮バイト数を指定します。

たとえば、デフォルト値 **4096** を使用すると、BIG-IP によって接続を保持して **Content-Length** ヘッダを書き換えるかどうかが決定される前に、最大 4096 バイトの圧縮データがバッファに格納されます。

BIG-IP が **Content-Length** ヘッダを書き換えるかどうかを決定する場合、（**[Response Chunking]** プロファイル設定を使用して）応答チャンクが有効になっているかどうかに左右されます。表 6.6 に、圧縮バッファサイズおよび応答チャンクに関する BIG-IP の動作を示します。

圧縮された応答のサイズ	圧縮された応答がチャンクされているかどうか	BIG-IP の処理
最大バッファサイズ以上	チャンクされている	接続を開いたままにします（**Connection** ヘッダが **[Close]** に設定されていない場合）。
	チャンクされていない	**Connection** ヘッダの値を **[Close]** に変更することで、接続を閉じます。
最大バッファサイズより小さい	チャンクされている	圧縮応答サイズを持つ **Content-Length** ヘッダを挿入しません。
	チャンクされていない	圧縮応答サイズを持つ **Content-Length** ヘッダを挿入します。

***表 6.6** 最大バッファサイズに基づいた BIG-IP の動作*

詳細については、*圧縮に対する最小コンテンツ長の指定*を参照してください。

圧縮バッファサイズを指定するには、**[Compression Buffer Size]** 設定で数値を入力します。

圧縮レベルの指定

[gzip Compression Level] 設定を使用して、データの圧縮率および圧縮速度を指定できます。指定できる値は、**[1 - Least Compression (Fastest)]**、**[9 - Most Compression (Slowest)]**、および **[Other]** です。デフォルトの圧縮レベルは、**[1 - Least Compression (Fastest)]** です。

1 から **9** までの任意の整数を指定できます。結果は次のとおりです。

- 指定する値が小さいほど、データの圧縮率は下がり、パフォーマンス速度は上がります。したがって、値 **1** を指定すると、最小の圧縮率、最高のパフォーマンスになります。
- 指定する値が大きいほど、データの圧縮率は上がり、パフォーマンス速度は下がります。したがって、値 **9**（指定可能な最大値）を指定すると、最高の圧縮率、最低のパフォーマンスになります。

◆ 警告

***[1 - Least Compression (Fastest)]** 以外の値を選択すると、システムパフォーマンスが劣化する可能性があります。*

◆ ヒント

*データ圧縮のための gzip レベルの使用法を変更することが可能です。これを行うには圧縮戦略を設定します。**圧縮戦略**とは、HTTP 応答に使用する圧縮プロバイダ（ハードウェアまたはソフトウェア）を特定*

*するものであり、利用可能な戦略には、Speed（デフォルト戦略）、Size、Ratio、Adaptive があります。圧縮戦略の設定方法については、「**データ圧縮戦略の使い方**」（6-22 ページ）を参照してください。*

gzip 圧縮のメモリレベルの指定

[**gzip Memory Level**] 設定では、**gzip** または **deflate** 圧縮方式を使用する場合に、BIG-IP がデータの圧縮に使用するメモリの量をキロバイト単位で指定します。[**gzip Memory Level**] 設定の値は、2 のべき乗の整数で、**1** ～ **256** の範囲のバイト数である必要があります。

一般に、指定する値が大きいほど、BIG-IP でより多くのメモリが使用され、圧縮速度および圧縮率が上がります。逆に、指定する値が小さいほど、BIG-IP で使用されるメモリは少なくなり、圧縮速度および圧縮率が下がります。デフォルト値は **8** です。

メモリレベルを指定するには、[**gzip Memory Level**] 設定で数値を選択します。

gzip 圧縮のウィンドウサイズの指定

[**gzip Window Size**] 設定では、**gzip** または **deflate** 圧縮方式を使用してサーバ応答を圧縮する場合に、BIG-IP が使用するウィンドウサイズの値をキロバイト単位で指定します。[**gzip Window Size**] 設定の値は、2のべき乗の整数で、**1**～**128**の範囲のバイト数である必要があります。

一般に、指定する値が大きいほど、BIG-IP でより多くのメモリが使用され、圧縮率が上がります。反対に、指定する値が小さいほど、BIG-IP で使用されるメモリは少なくなり、圧縮率が下がります。デフォルト値は **16** です。

ウィンドウサイズを指定するには、[**gzip Window Size**] 設定で数値を選択します。

Vary ヘッダの有効化または無効化

圧縮が有効になっている場合、[**Vary Header**] 設定によって、圧縮されたサーバ応答に **Vary: Accept-Encoding** ヘッダが挿入されます。応答内に **Vary** ヘッダがすでに存在する場合は、BIG-IP によって値 **Accept-Encoding** がそのヘッダに付加されます。

Vary: Accept-Encoding ヘッダをサーバ応答に挿入する理由は、RFC2616 による推奨事項（サーバ駆動型ネゴシエーションを受けるキャッシュ可能な応答のすべてに **Vary** ヘッダを挿入するべきである）に従うためです。HTTP 圧縮が適用されるサーバ応答は、このカテゴリに分類されます。

[**Vary Header**] 設定が無効になっている場合、BIG-IP は **Vary** ヘッダをサーバ応答に挿入しません。

Vary ヘッダを無効にするには、[**Vary Header**] 設定で [**Enabled**] ボックスをクリアします。

HTTP/1.0 要求に対する圧縮の許可

[**HTTP/1.0 Requests**] 設定は、下位互換性のために含まれており、HTTP/1.0 クライアントリクエストへの応答に対して HTTP 圧縮を許可します。この設定のデフォルト値は [**Disabled**] です。

この設定が [**Enabled**] に設定されている場合、次のいずれかの状況のみで、BIG-IP で応答が圧縮されます。

- サーバが **Connection: close** ヘッダで応答する
- 応答コンテンツが [**Compression Buffer Size**] 設定の値より大きくない

HTTP/1.0 要求に対して圧縮を有効にするには、[**HTTP/1.0 Requests**] 設定でボックスをチェックします。

Accept-Encoding ヘッダの保持

通常は、HTTP 圧縮を有効にすると、BIG-IP によって HTTP リクエストから **Accept-Encoding** ヘッダが除去されます。これにより、BIG-IP では、ターゲットサーバの代わりに HTTP 圧縮を実行できるようになります。

デフォルトでは、[**Keep Accept Encoding**] 設定は無効になっています。BIG-IP ではなくターゲットサーバで HTTP 圧縮を実行可能にするには、この設定を有効にします。

ブラウザの回避策の実装

[**Browser Workarounds**] 設定を有効にすると、コンテンツの圧縮時に発生する一般的なブラウザ問題に対する組み込みの回避策が BIG-IP で使用されます。デフォルト設定は無効になっています（解除）。具体的に言うと、この設定を有効にすると、次の条件のいずれかが発生した場合、BIG-IP はサーバ応答を圧縮しません。

- クライアントブラウザが Netscape® バージョン 4.0x である。
- クライアントブラウザが Netscape バージョン 4.x（つまり、バージョン 4.10 以上）であり、サーバ応答の **Content-Type** ヘッダが **text/html** または **text/plain** に設定されていない。
- クライアントブラウザが Microsoft® Internet Explorer®（任意のバージョン）であり、サーバ応答の **Content-Type** ヘッダが **text/css** または **application/x-javascript** のいずれかに設定されており、クライアント接続で SSL が使用されている。
- クライアントブラウザが Microsoft® Internet Explorer®（任意のバージョン）であり、サーバ応答の **Content-Type** ヘッダが **text/css** または **application/x-javascript** のいずれかに設定されており、サーバ応答の **Cache-Control** ヘッダが **no-cache** に設定されている。

CPU Saver

[**CPU Saver**] 設定を有効にすると、CPU 使用率が監視され、CPU 使用率が [**CPU Saver High Threshold**] または [**CPU Saver Low Threshold**] のいずれかで定義されているパーセンテージに達すると、圧縮速度が自動的に調整されます。デフォルトでは有効に設定されています (チェックされています)。

CPU Saver High Threshold

[**CPU Saver High Threshold**] 設定では、圧縮対象のコンテンツ量および適用される圧縮量が自動的に低減され始める CPU 使用率を指定します。デフォルト設定は **90** パーセントです。

CPU Saver Low Threshold

[**CPU Saver Low Threshold**] では、ユーザ定義の速度でコンテンツ圧縮が再開される CPU 使用率を指定します。デフォルト設定は **75** パーセントです。

データ圧縮戦略の使い方

標準のHTTPプロファイル設定プロバイダよりもさらに細かくデータ圧縮方法を制御したい場合は、デフォルト戦略以外の圧縮戦略を有効にします。デフォルトの圧縮戦略は [Speed] です。

本項では、BIG-IP で利用可能な圧縮プロバイダ、4 種類の圧縮戦略、および Adaptive 圧縮戦略の詳しい使い方について説明します。

◆注

圧縮戦略の設定に使用できるツールは BIG-IP コマンドラインインターフェイスのみとなっています。戦略を明示的に設定しない場合、あるいはコマンドラインインターフェイスではなく設定ユーティリティを使って圧縮を設定する場合は、Speed がデフォルト戦略として使用されます。

圧縮プロバイダの概要

BIG-IP では、圧縮プロバイダを利用して HTTP サーバ応答の圧縮を行います。***圧縮プロバイダ***はハードウェアカードもしくはソフトウェアプログラムの形態をとり、これらをマルチプロセッサ BIG-IP システムにインストールして、HTTP データの圧縮を実行できるようになっています。BIG-IP でのデータ圧縮に使用可能なソフトウェアおよびハードウェアの圧縮プロバイダには以下のものがあります。

- **ハードウェア圧縮カード**
  ハードウェア圧縮カードを使うことで、データ圧縮の速度が向上します。ハードウェア圧縮カードは、6400、6800、8400 の BIG-IP プラットフォームでオプションとして提供されており、8800 プラットフォームには標準搭載されています。
- **zlib**
  zlib ツールは、BIG-IP システムソフトウェアに含まれるソフトウェア圧縮プロバイダです。ハードウェア圧縮カードがビジー状態でそれ以上データを圧縮できない場合にのみ、**zlib** が使用されるようになっています。

ハードウェア圧縮プロバイダは、ソフトウェア圧縮プロバイダにくらべて圧縮品質のレベルが低くなっているため、注意が必要です。これに対して、ソフトウェア圧縮プロバイダでは、質の高い圧縮を提供するためにハードウェア圧縮プロバイダよりも多くのシステムリソースを必要とします。

bigpipe ユーティリティを使って、システムに配置されたハードウェア圧縮プロバイダを確認することができます。

圧縮戦略の選択について

コマンドラインインターフェイスを使って BIG-IP の圧縮設定を行う場合、Speed、Size、Ratio、Adaptive の 4 つの圧縮戦略から 1 つを選択することができます。BIG-IP では、ユーザが選択した***圧縮戦略***を使用して、どの圧縮プロバイダを使って任意の HTTP 応答を処理する

かを決定します。HTTP 応答はいったん圧縮プロバイダに割り当てられると、応答が完了するまでその圧縮プロバイダに関連付けられたままとなります。

表 6.7 で、4 つの圧縮戦略について説明しています。

圧縮戦略	説明
Speed	デフォルトの圧縮戦略です。 可能なかぎりハードウェア圧縮プロバイダが利用され、ハードウェアがビジー状態の場合にのみ、ソフトウェア圧縮プロバイダを使って HTTP サーバ応答の圧縮が行なわれます。**Speed** は、一括圧縮を行う場合や CPU オーバーヘッドを制限する場合にもっとも適した圧縮戦略です。
Size	BIG-IP は、TMM と Offload の比率を使い、できるだけ多くの圧縮をこのソフトウェアで実行します。ソフトウェアがビジー状態の場合にのみ、ハードウェア圧縮プロバイダを使って HTTP サーバ応答の圧縮が行なわれます。**Size** 戦略を使用することで、CPU オーバーヘッドの負担が軽減されます。
Ratio	重み付けラウンドロビン法を使って、データ圧縮に使用する圧縮プロバイダを決定します。Ratio 戦略を使用した場合、CPU オーバーヘッドが制限されると同時に、圧縮率が上がります。
Adaptive	BIG-IP では、最初にソフトウェア圧縮プロバイダを使って HTTP サーバ応答の圧縮を行います。HTTP プロファイルで設定した gzip 圧縮レベルと、システムに実装されたハードウェア圧縮プロバイダの両方に基づき、ハードウェア圧縮プロバイダへの切り替えが行われます。 システムの負荷が上昇すると、BIG-IP は希望の gzip 圧縮レベル（HTTP プロファイルで指定）を下げて対応します。ハードウェア圧縮プロバイダの利用は、ユーザが指定した gzip 圧縮レベル、もしくは系統的に下げられた gzip 圧縮レベルがそのプロバイダで提供可能な場合にかぎられます。 Adaptive 戦略を使えば、BIG-IP での圧縮処理を細かく制御することができます。

表 6.7 圧縮戦略の説明

Adaptive 圧縮戦略の概要

Adaptive 圧縮戦略では、BIG-IP がソフトウェアとハードウェアの両方の圧縮プロバイダを可能なかぎり効率的な方法で利用して、最高品質の圧縮を提供しつつ、ロードバランシングのためのリソースを確保できるように指定します。Adaptive 圧縮を有効にすると、BIG-IP は指定した品質レベル（gzip レベル）に基づき、有効な圧縮プロバイダから 1 つを選択して HTTP サーバ応答を圧縮します。Adaptive 圧縮は、ハードウェア圧縮カードが実装された BIG-IP 6400、6800、8400 でもっとも有効に機能します。

Adaptive 圧縮を使用するには、次のタスクを実行します。

- BIG-IP で Adaptive 圧縮を有効にする。
- HTTP プロファイルを作成して、gzip 圧縮レベルを設定する。

Adaptive 圧縮設定の詳細については、「*Adaptive 圧縮の設定*」（6-26 ページ）を参照してください。

Adaptive 圧縮戦略を設定すると、以下のような効果があります。

◆ システムの負荷が低下すると、Adaptive 圧縮戦略によってサーバ応答の圧縮品質を段階的に上げることができます。

- システムの負荷が上昇すると、Adaptive圧縮戦略によってサーバ応答の圧縮品質を段階的に下げることができます。これによって、増加したトラフィックのロードバランシングにシステムリソースをあてることが可能です。
- トラフィック量がピークに達した場合、およびHTTPプロファイルで設定した gzip 圧縮レベルに基づき、ハードウェア圧縮プロバイダを使ってデータの圧縮が開始されます。

プラットフォームごとの違いについて

Adaptive 圧縮戦略を設定する場合、BIG-IP で使用される圧縮プロバイダは、ユーザが設定した gzip レベル（圧縮の品質と速度の望ましいバランス）だけでなく、プラットフォームの種類によっても変わってきます。

表 6.8 では、各プラットフォームがデータ圧縮に使用する圧縮プロバイダについて説明しています。

BIG-IP プラットフォーム	使用される圧縮プロバイダ
6400/6800	ハードウェア圧縮カードがインストールされていれば、そのハードウェアカードが使用されます。ハードウェアカードがビジー状態であったり、ハードウェアカードがインストールされていない場合には、**zlib** が使われます。
8400	ハードウェア圧縮カードがインストールされていれば、ユーザが設定した gzip レベルに応じて、そのハードウェアカードか **zlib** のいずれかが使用されます。ハードウェアカードがインストールされていない場合は、**zlib** が使われます。
8800	設定された gzip レベルに応じて、ハードウェアカードか **zlib** のいずれかが使われます。ハードウェアカードがビジー状態の場合は、**zlib** が使われます。

***表 6.8** BIG-IP プラットフォームで使用される圧縮プロバイダ*

gzip レベルの概要

Adaptive 圧縮の仕組みを理解するには、HTTP プロファイルでの gzip 圧縮レベル設定について把握する必要があります。設定ユーティリティを使った HTTP プロファイルの設定（gzip 圧縮レベルの設定を含む）については、『***Configuration Guide for BIG-IP® Local Traffic Management***』を参照してください。

Adaptive 圧縮を有効にすると、ユーザが HTTP プロファイルで設定した gzip 圧縮レベルが使用されますが、その際の使用法はシステムにどのハードウェア圧縮プロバイダが実装されているかによって違ってきます。HTTP プロファイルの作成時に、gzip 圧縮レベルを **9** ～ **0** の範囲で設定します。gzip 圧縮レベルが高ければ高いほど、圧縮品質も上昇し、決められた品質に到達するためにより多くのリソースが必要になります。

たとえば、BIG-IP の RAM キャッシュ機能を利用して応答データを格納している場合は、gzip 圧縮レベルを **9** に設定します。これは、RAM キャッシュに格納されるデータは応答で継続的に再利用されることから、それらのデータの圧縮できわめて高い品質が求められるためです。

BIG-IP でのトラフィックフローが増加すると、圧縮品質はプロファイルで設定した gzip 圧縮レベルから自動的に低下します。指定された圧縮レベルにハードウェア圧縮プロバイダで対応できるポイントにまで gzip 圧縮レベルが低下すると、BIG-IP はソフトウェア圧縮プロバイダではなくハードウェア圧縮プロバイダを使用して、HTTP サーバ応答の圧縮を行います。

BIG-IP 6400、6800、8400 での gzip レベルの指定

ハードウェアカードが実装された BIG-IP 6400、6800、8400 で Adaptive 圧縮を有効にすると、そのシステムでの圧縮が HTTP プロファイルの gzip 圧縮レベル設定に影響を受けることになります。

表 6.9（6-25 ページ）に、6400、6800、8400 プラットフォームにおける gzip 圧縮レベルの影響についてまとめています。

gzip レベル	圧縮プロバイダへの影響
9 ～ 2	ソフトウェア圧縮プロバイダは、指定したレベルとシステム負荷の両方に基づき、圧縮を実行します。ソフトウェア圧縮プロバイダによって処理されるタスクの数が一定値（bigdb キー「**Compression.ProviderBusy**」により定義）を超えると、ハードウェア圧縮プロバイダが応答の圧縮を開始します。負荷が減り、利用可能なリソースが増えれば、ソフトウェア圧縮プロバイダがふたたび応答の圧縮を開始します。
1	bigdb 設定キー「**Compression.Adaptive.AHA.UseAtGzip1**」が有効になっていないかぎり、BIG-IP はソフトウェア圧縮プロバイダをハードウェア圧縮プロバイダより先に使用しようとします。このキーが有効になっていれば、ハードウェア圧縮プロバイダが使用されます。
0	BIG-IP は、ハードウェア圧縮プロバイダをソフトウェア圧縮プロバイダより先に使用しようとします。また、ハードウェア圧縮プロバイダ使用時には、最小限の圧縮が実行されます。 ***注**: gzip レベルを **0** に設定した場合、ある程度の圧縮が常に実行されることになります。*

***表 6.9** 6400、6800、8400 システムにおける gzip レベルの影響*

BIG-IP 8800 での gzip レベルの指定

ハードウェアカードが実装された BIG-IP 8800 で Adaptive 圧縮を有効にすると、同システムでの圧縮が HTTP プロファイルの gzip 圧縮レベル設定に影響を受けることになります。

表 6.10（6-26 ページ）に、8800 プラットフォームにおける gzip 圧縮レベルの影響についてまとめています。

gzip レベル	圧縮プロバイダへの影響
9 ～ 4	gzip 圧縮レベルを **9 ～ 4** に設定すると、システム負荷が上昇し始めるまで、ソフトウェア圧縮プロバイダが圧縮を実行します。システム負荷の上昇が始まると、それに従い gzip レベル（つまり圧縮品質）が低下します。 ソフトウェア圧縮プロバイダによって処理されるタスクの数が一定値（bigdb キー「**Compression.ProviderBusy**」により定義）を超えると、ハードウェア圧縮プロバイダが応答の圧縮を開始します。ソフトウェア圧縮プロバイダによって処理されるタスクの数がその値を下回った時点で、ソフトウェア圧縮プロバイダが応答の圧縮を再開します。
3 ～ 0	gzip 圧縮レベルを **3 ～ 0** に設定すると、タスクの数が一定値（bigdb キー「**Compression.ProviderBusy**」により定義）を超えるまで、ハードウェア圧縮プロバイダが圧縮を実行します。その後、ソフトウェア圧縮プロバイダが応答の圧縮を開始します。ハードウェア圧縮プロバイダによって処理されるタスクの数がその値を下回った時点で、ハードウェア圧縮プロバイダが応答の圧縮を再開します。 ***注**：ハードウェアプロバイダは **1** と **0** の gzip レベルをサポートしていません。したがって、**1** または **0** を指定したとしても、ハードウェアプロバイダは実際には **2** よりもわずかに上のレベルでデータの圧縮を行います。ハードウェア圧縮プロバイダがビジー状態になった場合は、ソフトウェアプロバイダがレベル **1** または **0** で圧縮を行うことができます。*

***表 6.10** 8800 システムにおける gzip レベルの影響*

Adaptive 圧縮の設定

Adaptive 圧縮の設定では、カスタム HTTP プロファイルの圧縮設定だけでなく、コマンドラインインターフェイスもあわせて使用します。コマンドラインインターフェイスを使用するのは、圧縮戦略を Speed（デフォルト）から Adaptive に変更する場合です。

Adaptive 圧縮戦略では、HTTP プロファイルで指定した [**gzip Compression Level**] 設定が使用されます。gzip レベルと [**gzip Compression Level**] 設定の詳細については、「*gzip レベルの概要*」（6-24 ページ）と「*圧縮レベルの指定*」（6-18 ページ）を参照してください。

圧縮統計情報の表示

BIG-IP の圧縮計情報を表示するには、**bigpipe** ユーティリティを使用します。トラフィックスループットの情報のほか、ハードウェア圧縮プロバイダごとの合計圧縮率もあわせて確認することが可能です。

表 6.11（6-26 ページ）で、表示される圧縮統計情報について説明しています。

圧縮統計情報	説明
pre	BIG-IP が受信した圧縮されていないトラフィックの量。
post	BIG-IP が送信した圧縮されたトラフィックの量。

***表 6.11** 圧縮統計情報の説明*

圧縮統計情報	説明
null	BIG-IP が圧縮ヘッダにラップした圧縮されていないコンテンツの量。 ***注：****BIG-IP はコンテンツを圧縮ヘッダにラップしますが、許可された圧縮量（メガバイト）を超えた場合、もしくは CPU サーバがアクティブになっている場合は、それらのコンテンツが圧縮されることはありません。*
saved	圧縮前のコンテンツ量と圧縮後のコンテンツ量の比率。

表 6.11 *圧縮統計情報の説明（続き）*

図 6.1 は **bigpipe** コマンド **http show** コマンドの結果例です。圧縮統計情報は出力の最終行に示されます。

```
GLOBAL HTTP STATISTICS
|    requests (total, max in conn, GET, POST) = (0, 0, 0, 0)
|    requests (v0.9, v1.0, v1.1) = (0, 0, 0)
|    responses (v0.9, v1.0, v1.1) = (0, 0, 0)
|    responses (2xx, 3xx, 4xx, 5xx) = (0, 0, 0, 0)
|    response size (   <1k,   1-4k,  4-16k, 16-32k, 32-64k)
|             (     0,      0,      0,      0,      0)
|    Set-Cookie header insertions = 0
|    RAM cache (entries, size, evict) = (0, 0, 0)
|    RAM cache hit count (hits, misses, misses all) = (0, 0, 0)
|    RAM cache byte count (hits, misses, misses all) = (0, 0, 0)
COMPRESSION STATISTICS --
|    total bytes (pre, post, null, saved) = (0, 0, 0, 0.00%)
```

図 6.1 *BIG-IP 8800 に対する* ***http show*** *コマンドの出力例*

RAM キャッシュ機能の設定

この項では、BIG-IP で RAM キャッシュ機能のプロパティを設定する方法について説明します。***RAM キャッシュ***とは、BIG-IP の RAM に格納されている HTTP オブジェクトのキャッシュで、バックエンド サーバの負荷を軽減するために、以降の接続で再利用されるものです。

RAM キャッシュの基礎

BIG-IP で RAM キャッシュ機能を設定する前に、以下のような概念を考慮に入れる必要があります。

- どのような場合に RAM キャッシュを使用するか
- どのような項目がキャッシュ可能か
- キャッシュのメカニズム

どのような場合に RAM キャッシュ機能を使用するか

RAM キャッシュ機能を使用すると、バックエンドサーバへのトラフィック負荷を軽減できます。サイトのオブジェクトに対する需要が高い場合、サイトに大量のスタティックコンテンツがある場合、またはサイトのオブジェクトが圧縮されている場合に、この機能は役立ちます。

- **高需要オブジェクト**
  この機能は、特定のコンテンツに対する需要が高い時間帯があるサイトに役立ちます。RAM キャッシュが設定されている場合、コンテンツサーバは有効期限ごとに 1 度だけ、BIG-IP にコンテンツを配信する必要があります。
- **スタティックコンテンツ**
  サイトが大量のスタティックコンテンツ（CSS ファイル、JavaScript ファイル、イメージ、ロゴなど）で構成されている場合にも、この機能が役立ちます。
- **コンテンツ圧縮**
  圧縮可能なデータについては、RAM キャッシュでは圧縮データの受け付けが可能なクライアント用にデータを格納できます。RAM キャッシュを BIG-IP の圧縮機能と併用した場合、BIG-IP およびコンテンツサーバの負荷が軽減されます。

キャッシュ可能な項目

RAM キャッシュ機能は、RFC 2616、*Hypertext Transfer Protocol -- HTTP/1.1* に記述されているキャッシュ仕様に完全に準拠しています。つまり、次のコンテンツタイプをキャッシュするように RAM キャッシュ機能を設定できます。

- **200**、**203**、**206**、**300**、**301**、および **410** 応答
- **GET** メソッドへの応答（デフォルト）
- キャッシュコンテンツで包含が指定されているか、iRule で指定されている URI に対するその他の HTTP メソッド

- **User-Agent** 値および **Accept-Encoding** 値に基づくコンテンツ。RAM キャッシュでは、**Vary** ヘッダに対して異なるコンテンツが保持されます。

RAM キャッシュでキャッシュされない項目は、次のとおりです。

- キャッシュコントロールヘッダで指定されたプライベートデータ
- **HEAD**、**PUT**、**DELETE**、**TRACE** および **CONNECT** メソッド（デフォルト）

RAM キャッシュメカニズムについて

デフォルトの RAM キャッシュ設定では、HTTP **GET** メソッドへの応答のみがキャッシュされます。ただし、HTTP 以外のメソッドを含む他のメソッドをキャッシュする場合でも、RAM キャッシュを使用できます。これは、HTTP プロファイル内の URI **Include** または **Pin** リストの URI を指定するか、iRule を作成することで実行できます。

この項の残りの部分では、次の質問に回答します。

- BIG-IP が実際にキャッシュするのはどの応答ですか。
- HTTP プロファイルの **Include リスト**に URI を追加するとどうなりますか。
- **Pin List**、**Include List**、および **Exclude リスト**を処理する場合、BIG-IP が使用する優先順位はどのようになっていますか。

BIG-IP がキャッシュするのはどの応答ですか。

BIG-IP は、HTTP リクエストとその応答を評価して、どの応答をキャッシュするかを判断します。表 6.12 に、応答をキャッシュに包含するか、除外するかを決定する前に、BIG-IP が使用する要求と応答内の基準を示します。

基準	アクション	タイミング
HTTP リクエストの基準		
BIG-IP が、**HTTP_REQUEST** イベント内で **CACHE::disable** コマンドを呼び出す。	除外	常時
要求が、HTTP プロファイルの **Exclude** リスト の項目に合致する。	除外	常時
要求が、HTTP プロファイルの **Pin** リストの項目に合致する。	包含	常時
要求が、HTTP プロファイルの **Include** リストの項目に合致する。	包含	常時
要求が、標準外の HTTP メソッドを含む、**GET** 以外のメソッドを指定する。	除外	常時。ただし、**Pin** または **Include** リストの項目か、iRule でオーバーライドされる場合を除く。
BIG-IP が、**CACHE_REQUEST** イベントでトリガされた iRule を処理する。	包含	常時

***表 6.12** 応答をキャッシュするかどうかを決定する要求および応答の基準*

基準	アクション	タイミング
HTTP 応答の基準		
Vary ヘッダの値が、**User-Agent** や **Accept-Encoding** 要求ヘッダのみで構成される。	1つのURLに複数の応答を含める	常時
Vary ヘッダの値が、同じリソースの後続の応答で変化する。	既存のキャッシュエントリを無効化する	常時
Vary ヘッダの値が、他の要求ヘッダや特殊な値 ***** （アスタリスク）で構成される。	除外	常時
応答が、次以外のステータスコードを含む。 **200** (OK) **203** (Non-Authoritative Information) **206** (Partial Content) **300** (Multiple Choices) **301** (Permanent Redirect) **410** (Gone)	除外	常時
応答の **Content-Length** ヘッダの値が **0** である。	除外	常時。ただし、応答が **Content-Length** ヘッダを含まない場合を除く（すなわち、**Transfer-Encoding** がチャンクされているか、応答がサーバの接続終了に依存している場合）。
応答のオブジェクトサイズが、設定された応答の最大および最小オブジェクトサイズと最大キャッシュサイズを超えている。 ***注*: *オブジェクトサイズという用語は、応答ヘッダと応答の本文の両方を示します。***	除外	オブジェクトサイズが最大キャッシュサイズを超えている場合は、常時。 そうでない場合、**Pin** または **Include** リストの項目か、iRule でオーバーライドされる場合を除き、常時。
応答が、無効な値を持つ **Expires** ヘッダを含む。	除外	常時。ただし、**Pin** または **Include** リストの項目か、iRule でオーバーライドされる場合を除く。
要求が、クエリ文字列を含むが、応答に **Expires** ヘッダが含まれない。	除外	常時。ただし、**Pin** または **Include** リストの項目か、iRule でオーバーライドされる場合を除く。
応答が、**no-store**、**no-cache**、または **private** の値を持つ **Cache-Control** ヘッダを含む。	除外	常時。ただし、**Pin** または **Include** リストの項目か、iRule でオーバーライドされる場合を除く。
要求が、**Authorization** ヘッダを含み、応答の **Cache-Control** ヘッダに **s-maxage**、**must-revalidate**、**public** の値がない。	除外	常時。ただし、**Pin** または **Include** リストの項目か、iRule でオーバーライドされる場合を除く。

***表 6.12** 応答をキャッシュするかどうかを決定する要求および応答の基準（続き）*

HTTP プロファイルの Include リストに URI を追加するとどうなりますか。

HTTP プロファイルの **Include リスト**に含まれる URI の場合、BIG-IP は、すべての要求メソッドで応答をキャッシュします。その他の制約条件もすべて適用されます。

BIG-IP が使用する優先順位はどのようになっていますか。

URL リストを使用して、BIG-IP でキャッシュする応答を決定する場合、BIG-IP では、次の優先順位を降順で使用します。

- **Exclude リスト**
- **Pin リスト**
- **Include リスト**

コンテンツに基づいた BIG-IP のアクションについて

BIG-IP は、コンテンツのタイプに応じて、キャッシュコンテンツに何らかのアクションを実行します。表 6.13 は、これらのアクションを示します。

アクション	キャッシュコンテンツのタイプ
削除	BIG-IP は、キャッシュコンテンツからすべての Cookie ヘッダを削除します。
変更	BIG-IP は、接続時にホップバイホップヘッダを削除します。これらのヘッダには、**Connection**、**Keep-Alive**、および **Transfer Encoding** があります。
追加	BIG-IP の現在の時間を含む **Date** ヘッダを追加します。 また、項目がキャッシュに格納されている時間を反映する **Age** ヘッダも追加します。この設定は、HTTP プロファイルではデフォルトで有効になっています。プロファイルで [**Insert Age Header**] を無効にすると、この設定を無効化できます。
そのまま格納	BIG-IP は、その他すべてのヘッダを受け取ったまま、変更せずに格納します。

***表 6.13** ヘッダタイプに基づいた RAM キャッシュアクション*

RAM キャッシュからの項目の消去

RAM キャッシュでは、キャッシュ内で使用頻度の低い項目が削除されます。これにより、新しい項目をキャッシュするよう選択した場合に、キャッシュ内のスペースが古い項目で占有されることを防ぎます。また、キャッシュではスコアリングシステムを使用して、一定期間の後に古い項目を削除します。キャッシュされた項目がこの経過時間制限に達すると、期限切れになり、キャッシュから削除されます。HTTP プロファイルの属性を使用すると、各キャッシュインスタンスのサイズや、BIG-IP がキャッシュから期限切れ項目を削除する期間を制御できます。これらの属性の詳細については、次の項「*RAM キャッシュ設定について*」(6-32 ページ）を参照してください。

RAM キャッシュ設定について

RAM キャッシュを設定するには、HTTP プロファイルの RAM キャッシュ設定を行う必要があります。これらの設定では、キャッシュをオンにしたり、特定の実装に対してキャッシュを微調整することが可能です。HTTP プロファイル内で RAM キャッシュオブジェクトを設定する以外に、**bigpipe** ユーティリティコマンドを使用したり、**bigdb** 設定キーを設定したり、iRules を作成することができます。RAM キャッシュ機能に関連した **bigdb** キーについては、『***Bigpipe Utility Reference Guide***』の付録 B を参照してください。RAM キャッシュに iRule™ を作成する詳細については、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。

キャッシュに格納されるデフォルトの項目は、**HTTP GET** 応答です。ただし、特定の URI に対して **POST** メソッドおよび **GET** メソッドをキャッシュする場合は、URI リスト内で URI を指定できます。

RAM キャッシュは以下のいずれかの方法で実装することができます。

- デフォルトの **http** プロファイルをもとにカスタムプロファイルを作成し、RAM キャッシュ設定を変更します。
- **http-lan-optimized-caching** プロファイルをそのまま実装します。このプロファイルは、RAM キャッシュを最適化するように最初から設定されています。
- **http-lan-optimized-caching** プロファイルをもとにカスタムプロファイルを作成し、RAM キャッシュ設定を変更します。

詳細については、「*http-lan-optimized-caching プロファイルの使い方*」（6-36 ページ）と次の項の「*RAM キャッシュへの HTTP プロファイルの使用*」を参照してください。

RAM キャッシュへの HTTP プロファイルの使用

前述したように、RAM キャッシュを実装するひとつの方法が、デフォルトの **http** プロファイルに基づいてカスタムプロファイルを作成することです。表 6.14 に、作成したカスタム HTTP プロファイル内で指定可能な RAM キャッシュ設定を示します。これらの設定を行う場合は、デフォルト値が存在しない場合に値を指定するか、またはデフォルト値を変更します。

設定	説明	デフォルト値
RAM Cache	RAM キャッシュ機能を有効または無効に指定します。	**Disabled**
Maximum Cache Size	RAM キャッシュの最大サイズ（メガバイト単位）を指定します。キャッシュが最大サイズに達すると、最も古いエントリが削除され始めます。	**100**
Maximum Entries	RAMキャッシュに格納できるエントリの最大数を指定します。	**10000**

***表 6.14** HTTP プロファイルの RAM キャッシュ設定*

設定	説明	デフォルト値
Maximum Age	キャッシュコンテンツが有効とみなされる時間（秒単位）を指定します。	**3600**
Minimum Object Size	キャッシュ対象として適格とみなされる最小オブジェクト（バイト単位）を指定します。	**500**
Maximum Object Size	キャッシュ対象として適格とみなされる最大オブジェクト（バイト単位）を指定します。	**50000**
URI Caching	RAM キャッシュ内の特定の URI が BIG-IP で保持されるか除外されるかを指定します。このプロセスは、通常はキャッシュ対象として不適格な URI をキャッシュすること、または通常はキャッシュ対象として適格な URI をキャッシュしないことを BIG-IP に強制します。	**Not Configured**
URI List	BIG-IP がキャッシュに包含するかキャッシュから除外するユニフォームリソース識別子（URI）を指定します。 **Pin リスト** RAMキャッシュに無制限に格納させる応答のURIをリストします。 **Include List** 通常はキャッシュ対象として不適格で、BIG-IP によってキャッシュされる URI をリストします。Include List に URI を追加すると、BIG-IP は、HTTP 以外のメソッドを含め、**GET** メソッドおよびその他のメソッドをキャッシュします。 **Exclude List** 通常はキャッシュ対象として適格で、BIG-IP によってキャッシュされない URI をリストします。	デフォルト値なし
Ignore Headers	RAM キャッシュが有効になっている場合に、BIG-IP によってクライアント側の **Cache-Control** ヘッダが処理される方法を指定します。 **All** BIG-IP によってすべての **Cache-Control** ヘッダが無視されるように指定します。 **Cache-Control:max-age** **Cache-Control:max-age**ヘッダのみが無視されるように指定します。 **None** すべての **Cache-Control** ヘッダが受け入れられるように指定します。	**All**
Insert Age Header	有効になっている場合、**Date**ヘッダおよび**Age**ヘッダがキャッシュエントリに挿入されるように指定します。**Date** ヘッダには、BIG-IP 上の現在の日時が含まれます。**Age** ヘッダには、コンテンツがキャッシュに格納されている時間が含まれます。	**Enabled**
Aging Rate	キャッシュエントリがエージングされる速度を指定します。エージング速度の範囲は、0（最も遅いエージング）～ 10（最も速いエージング）です。	**9**

表 6.14 HTTP プロファイルの RAM キャッシュ設定（続き）

データ圧縮と RAM キャッシュの最適化

BIG-IP には、F5 がユーザ用に作成した一連のカスタム HTTP プロファイルが含まれています。これらのプロファイルは、データ圧縮と RAM キャッシュのパフォーマンスを最適化するためのものです。F5 により定義されたこれらのプロファイルは、**http** デフォルトプロファイルに基づいており、圧縮と RAM キャッシュの最適なパフォーマンスを確保するために事前設定されています。これらのプロファイルには以下のものがあります。

- **http-acceleration**
- **http-wan-optimized-compression**
- **http-lan-optimized-caching**
- **http-wan-optimized-compression-caching**

http-acceleration プロファイルの使い方

http-acceleration は HTTP タイププロファイルであり、BIG-IP でユーザ用に事前作成されたカスタムプロファイルとなっています。**http-acceleration** プロファイルを実装することで、カスタムプロファイルを作成しなくても、HTTP トラフィックを加速することができます。

場合によっては、別の BIG-IP システムモジュール（BIG-IP WebAccelerator）で **http-accleration** プロファイルを有効にすることも可能です。

http-acceleration プロファイルは **http** プロファイルの設定とそれらのデフォルト値を継承していますが、設定値の一部は変更されています。このプロファイルは、HTTP 高速化を最適化しつつ、ほかの設定には **http** プロファイルのデフォルト値を使いたいという場合に適しています。

http-accleration プロファイルをそのまま使用するか、もしくは **http-accleration** プロファイルを親プロファイルとして別のカスタムプロファイルを作成することも可能です。

http-acceleration プロファイルのデフォルト値は、ほとんどが **http** プロファイルのデフォルト値と同じですが、いくつか違うものもあり、それらを表 6.15 にまとめています。

設定	説明	デフォルト値
RAM Cache	RAM キャッシュ機能を有効または無効に指定します。	**Enabled**
Minimum Object Size	キャッシュ対象として適格とみなされる最小オブジェクト（バイト単位）を指定します。	**0**
Maximum Object Size	キャッシュ対象として適格とみなされる最大オブジェクト（バイト単位）を指定します。	**4194304**

表 6.15 http プロファイルのものと異なる http-acclerator プロファイルの値

http-wan-optimized-compression プロファイルの使い方

http-wan-optimized-compression はHTTP タイププロファイルであり、BIG-IP でユーザ用に事前作成されたカスタムプロファイルとなっています。**http-wan-optimized-compression** プロファイルを実装することで、カスタムプロファイルを作成しなくても、データ圧縮のパフォーマンスを最適化することができます。

http-wan-optimized-compression プロファイルは **http** プロファイルの設定とそれらのデフォルト値を継承していますが、設定値の一部は変更されています。このプロファイルは、データ圧縮を最適化しつつ、ほかの設定には **http** プロファイルのデフォルト値を使いたいという場合に適しています。

http-wan-optimized-compression プロファイルをそのまま使用するか、もしくは **http-wan-optimized-compression** プロファイルを親プロファイルとして別のカスタムプロファイルを作成することも可能です。

http-wan-optimized-compression プロファイルのデフォルト値は、ほとんどが **http** プロファイルのデフォルト値と同じですが、いくつか違うものもあり、それらを表 6.16 にまとめています。

設定	説明	デフォルト値
Compression	HTTP 圧縮機能を有効または無効にします。	**Enabled**
Compression Buffer Size	圧縮サイズを指定する **Content-Length** ヘッダを応答に挿入するかどうかが BIG-IP によって決定される前に、バッファに格納される最大圧縮バイト数を指定します。	**131072**
HTTP/1.0 Requests	HTTP/1.0 クライアントリクエストへの応答の圧縮を有効または無効にします。	**Enabled**
gzip Compression Level	圧縮の量とレートを指定します。	**1 - Least Compression (Fastest)**
gzip Memory Level	サーバ応答の圧縮時に、BIG-IP が内部圧縮バッファに使用するメモリをキロバイト単位で指定します。	**16**
gzip Window Size	サーバ応答の圧縮時に BIG-IP が使用するウィンドウサイズをキロバイト単位で指定します。	**64**
Vary Header	キャッシュ可能なサーバ応答への **Vary** ヘッダの挿入を有効または無効にします。	**Enabled**
Response Chunking	HTTP 応答に対するチャンクの処理方法を指定します。指定できる値は、[**Unchunk**]、[**Rechunk**]、[**Selective**]、および [**Preserve**] です。	**Selective**

表 6.16 http プロファイルのものと異なる http-wan-optimized-compression プロファイルの値

http-lan-optimized-caching プロファイルの使い方

http-lan-optimized-caching は HTTP タイププロファイルであり、BIG-IP でユーザ用に事前作成されたカスタムプロファイルとなっています。**http-lan-optimized-caching** プロファイルを実装することで、カスタムプロファイルを作成しなくても、RAM キャッシュのパフォーマンスを最適化することができます。

http-lan-optimized-caching プロファイルは **http** プロファイルの設定とそれらのデフォルト値を継承していますが、設定値の一部は変更されています。このプロファイルは、RAM キャッシュを最適化しつつ、ほかの設定には **http** プロファイルのデフォルト値を使いたいという場合に適しています。

http-lan-optimized-caching プロファイルをそのまま使用するか、もしくは **http-lan-optimized-caching** プロファイルを親プロファイルとして別のカスタムプロファイルを作成することも可能です。

http-lan-optimized-caching プロファイルのデフォルト値は、ほとんどが **http** プロファイルのデフォルト値と同じですが、いくつか違うものもあり、それらを表 6.17 にまとめています。

設定	説明	デフォルト値
RAM Cache	RAM キャッシュ機能を有効または無効に指定します。	**Enabled**
Maximum Cache Size	RAM キャッシュの最大サイズ（メガバイト単位）を指定します。キャッシュが最大サイズに達すると、最も古いエントリが削除され始めます。	**10**
Maximum Age	キャッシュコンテンツが有効とみなされる時間（秒単位）を指定します。	**86400**
Minimum Object Size	キャッシュ対象として適格とみなされる最小オブジェクト（バイト単位）を指定します。	**0**
Maximum Object Size	キャッシュ対象として適格とみなされる最大オブジェクト（バイト単位）を指定します。	**2000000**

表 6.17 http プロファイルのものと異なる http-lan-optimized-caching プロファイルの値

http-wan-optimized-compression-caching プロファイルの使い方

http-wan-optimized-compression-caching もまたHTTPプロファイルの一種であり、**http**、**http-wan-optimized-compression**、**http-lan-optimized-caching** の 3 つのプロファイルのデフォルト設定値を組み合わせたものとなっています。このプロファイルは、データ圧縮と RAM キャッシュの両方を最適化しつつ、ほかの設定には **http** プロファイルのデフォルト値を使いたいという場合に適しています。

このプロファイルのデフォルト設定値については、「*HTTP プロファイル設定について*」（6-2 ページ）、「*http-wan-optimized-compression プロファイルの使い方*」(6-35 ページ)、および「*http-lan-optimized-caching プロファイルの使い方*」（6-36 ページ）を参照してください。

FTP プロファイルの設定

BIG-IP には、File Transfer Protocol（FTP）トラフィックの管理に使用するためのプロファイルタイプが含まれます。FTP プロファイルの設定は、各自のニーズに合わせて調整できます。デフォルト値を持つ設定については、デフォルト設定をそのまま使用することも変更することもできます。設定の変更は、プロファイルの作成時に行うか、プロファイルの作成後にいつでも行うことができます。プロファイルの設定に関する手順については、第 5 章「*プロファイルについて*」を参照してください。

表 6.18 に、構成可能な設定およびそれぞれの概要とデフォルト値を列挙します。この表の後に、特定の設定について説明します。

汎用プロパティ	説明	デフォルト値
Name	プロファイルのユーザ定義名を指定します。プロファイル名の指定は必須です。	デフォルト値なし
Parent Profile	カスタムプロファイルの派生元とするプロファイルを指定します。	**ftp**
Translate Extended	FTP プロトコルを使用する際の IPv4 と IPv6 のクライアントおよびサーバ間の互換性を確保します。	**Enabled**
Data Port	FTP サービスを代替ポート上で実行可能にします。	**20**
Protocol Security	チェックされている（有効になっている）場合、BIG-IP® Protocol Security Module のセキュリティプロファイルを使って、FTP トラフィックのセキュリティ脆弱性をチェックします。このオプションは Protocol Security Module のライセンスがある場合にのみ利用可能です。	Disabled

***表 6.18** FTP プロファイルの設定*

FTP プロファイルを設定する前に、変更する可能性がある特定のノード設定に関する説明を読んでおくと役立ちます。

プロファイル名の指定

FTP プロファイルを作成するには、プロファイルの一意の名前を指定する必要があります。[**Name**] 設定は、FTP プロファイルの作成時に値をアクティブに指定する必要がある 2 つの設定のうちの 1 つです。その他の設定にはすべて、デフォルト値があります。

親プロファイルの指定

作成する各プロファイルは、親プロファイルから派生します。**Parent Profile** 設定で、デフォルトの **ftp** プロファイルを親プロファイルとして選択するか、すでに作成している別の FTP プロファイルを選択することができます。

Translate Extended 値の指定

（IPv4 のアドレスとは異なり、IPv6 のアドレスは 32 ビットに制限されていないため、IP バージョンが混合した構成で FTP を使用する場合、互換性の問題が生じることがあります。

[**Translate Extended**] 設定は、デフォルト で有効になっており 、IPv4 と IPv6 の両方のシステムがクライアント - サーバ構成に含まれる場合、BIG-IP によって FTP コマンド が自動的に変換されます。たとえば、IPv4を搭載したクライアント システムが IPv6を搭載したサーバに FTP **PASV** コマンド を送信した場合、この **PASV** コマンド は、IPv6 システムの同じ FTP コマンド に相当する **EPSV** に自動で変換されます。

EPRV と **PORT** の FTP コマンドについても同じように変換が行われます。

この設定のデフォルト値（[**Enabled**]）を変更する必要がある可能性はほとんどありません。この設定を無効にする必要があるのは、**EPSV** コマンドを IPv4 システムに送信する場合（FTP サーバのテスト時など）だけです。

データポートの指定

Data Port 設定を使用すると、FTP サービスを代替ポート上で実行できます。デフォルトのポート番号 **20** を使用するか、別のポート番号を指定できます。

FTP トラフィックのセキュリティの有効化

BIG-IP に BIG-IP Application Security Manager のライセンスが含まれる場合、アプリケーションセキュリティ設定の [Advanced Firewall] 機能を使って、FTP トラフィックのセキュリティスキャンを有効にすることができます。詳細については、『***Configuration Guide for BIG-IP® Application Security Management***』の「*Working with the Advanced Firewall*」を参照してください。

SIP プロファイルの設定

BIG-IP には、Session Initiation Protocol（SIP）トラフィックの管理に使用するためのサービスプロファイルが含まれます。***Session Initiation Protocol（SIP）***は、複数の参加者で構成されるセッションを管理するアプリケーションレイヤプロトコルであるため、リアルタイムのメッセージ、音声、データ、および映像が有効化されます。セッションは電話やインスタントメッセージによる単純な双方向の会話の場合もあれば、音声、データ、ビデオで構成される複雑でコラボレーティブなマルチメディアカンファレンスコールの場合もあります。

アプリケーションレベルセッションである SIP セッションは、3 つのレイヤ 4 プロトコル（SCTP、TCP、UDP）のいずれか 1 つを使って実行されます。SIP セッションをどのように処理するかは SIP プロファイルで設定します。指定されたレイヤ 4 プロトコルプロファイルを使い、必要なポートを解放して BIG-IP 経由でデータを伝送できるようにバーチャルサーバを設定します。バーチャルサーバに SIP プロファイルを割り当てる際に、SCTP、TCP、UDP のいずれかのプロファイルも一緒にサーバに割り当てることができます。これらのプロトコルプロファイルをサーバに割り当てない場合、BIG-IP によっていずれか 1 つが自動的に割り当てられるようになっています。プロトコルプロファイルの詳細については、第 8 章「*プロトコルプロファイルの管理*」を参照してください。

SIP プロファイルによって、BIG-IP が **Call-ID** を使って SIP セッションのパーシステンスを処理するように自動で設定されます。***Call-ID***は、アプリケーション間でやり取りされる一連のメッセージを 1 つのグループにまとめるグローバル一意識別子です。SIP セッションのパーシステンスが BIG-IP でどのように処理されるかを、カスタマイズできるようになっています。カスタマイズを行うには、SIP パーシステンスプロファイルを作成します。SIP パーシステンスプロファイルを使用するには、SIP プロファイルもあわせて使用する必要があり、SIP プロファイルと SIP パーシステンスプロファイルの両方を 1 つのバーチャルサーバに割り当てます。SIP パーシステンスの設定の詳細については、「*SIP パーシステンスのプロファイル設定について*」（7-17 ページ）を参照してください。

SIP プロファイル設定の概要

設定ユーティリティの [New SIP Profile] 画面では、SIP 設定が [General Properties] と [Settings] の 2 つのカテゴリに分けられています。SIP プロファイルの作成時には、設定をそのまま使用することも、変更することもできます。また、後日プロファイルを変更することも可能です。プロファイル設定の具体的な手順については、第 5 章「*プロファイルについて*」を参照してください。

表 6.19（6-41 ページ）では、SIP プロファイルで指定可能な設定を、それぞれの概要とデフォルト値と共に一覧にしています。この表の後に、SIP プロファイルの設定およびそれらの変更手順について詳しく説明しています。

汎用プロパティ	説明	デフォルト値
Name	プロファイルのユーザ定義名を指定します。この設定は必須です。	デフォルト値なし
Parent Profile	カスタムプロファイルの派生元とするプロファイルを指定します。	**sip**
Maximum Size (Bytes)	BIG-IP で対応可能な SIP メッセージの最大サイズを指定します。SIPメッセージのサイズがこの値を上回る場合は、接続が中断されます。	**65535 バイト**
Dialog Aware	SIP ダイアログ情報をスヌープして、それらを既知の SIP ダイアログに転送するように指定します。この設定を有効にすると、[**Community**] 設定が画面に表示されます。	Disabled（チェックなし）
Community	そのプロファイルが所属するプロキシ機能グループを示す文字列を指定します。この設定は、[**Dialog Aware**] 設定が有効になっている場合にのみ表示されます。	デフォルト値なし
Terminate on BYE	BYE トランザクション終了時の接続の切断を有効化 / 無効化します。***BYE トランザクション***とは、2 つのアプリケーション間の接続を閉じる準備が整った時点で、一方のアプリケーションが相手のアプリケーションに送信するメッセージのことです。BIG-IP が BYE トランザクションに遭遇した場合、[**Terminate on BYE**] 設定が有効になっていれば、接続が切断されます。このパラメータは UDP 接続でのみ使用されるものであり、TCP 接続では使用できません。	Enabled（チェックあり）
Insert Via Header	SIP 応答への **Via** ヘッダの挿入を有効化 / 無効化します。***Via*** ヘッダはメッセージの発信元を示すものであり、応答メッセージではこの経路情報が使用されます。	**Disabled**
User Via	[**Insert Via Header**] 設定を有効にする際に、SIP 要求の先頭の **Via** ヘッダの値として挿入する文字列を指定します。この値は、SIP プロトコルのほか、仮想アドレスとポートを指定するものです。	デフォルト値なし
Secure Via Header	SIP 応答への Secure Via ヘッダの挿入を有効化 / 無効化します。***Secure Via Header*** はメッセージの発信元を示すものであり、SSL/TLS と共に使用することで、システムを保護することができます。応答メッセージではこの経路情報が使用されます。	Disabled（チェックなし）
Insert Record-Route Header	後続の SIP 要求メッセージのネクストホップを示す Record-Route SIP ヘッダの挿入を有効化 / 無効化します。	Disabled（チェックなし）

表 6.19 *SIP プロファイルの設定*

SIP プロファイルの設定に取り掛かる前に、変更する可能性のある設定値についての説明を読んでおくと役立ちます。

プロファイル名の指定

SIP プロファイルを作成するには、プロファイルの一意の名前を指定する必要があります。[**Name**] は、SIP プロファイルの作成時に積極的に値を指定する必要がある唯一の設定です。その他の設定にはすべてデフォルト値があります。

プロファイル名を指定するには、[**Name**] 設定で、プロファイルの一意の名前を入力します。

親プロファイルの指定

作成する各プロファイルは、親プロファイルから派生します。デフォルトの **sip** プロファイルを親プロファイルとして使用するか、すでに作成している別の SIP プロファイルを使用することができます。

親プロファイルを指定するには、[**Parent Profile**] 設定でプロファイル名を選択します。

最大メッセージサイズの指定

BIG-IP は、**65535** バイト以下の着信 SIP メッセージに対応します。SIP メッセージのサイズがこの値を上回る場合は、接続が中断されます。

システムで許容可能な最大メッセージサイズを変更するには、[**Maximum Size (Bytes)**] ボックスに別の値を入力します。

ダイアログスヌーピングの有効化

[**Dialog Aware**] 設定を有効にすると、BIG-IP は SIP ダイアログ情報をスヌープして、SIP メッセージを既知の SIP ダイアログに自動で転送します。この設定を有効にすると、設定ユーティリティで [**Community**] 設定が表示されます。

コミュニティ文字列の指定

[**Community**] 設定で、プロキシ機能グループの名前を指定することができます。複数のバーチャルサーバで、それぞれが SIP タイププロファイルを参照する必要があり、それらのプロファイルの 2 つ以上を同じプロキシ機能グループに所属させたい場合に、この設定を使用します。この設定は、[**Dialog Aware**] 設定が有効になっている場合にのみ表示されます。

接続終了基準の指定

セッションを開始したアプリケーション（クライアント）もしくはそのセッションに応答したアプリケーション（サーバ）のどちらかが BYE トランザクションを発行した時点で、BIG-IP は SIP 接続を終了させます。これは、UDP で SIP セッションを実行している場合には適切に機能しますが、SCTP 接続もしくは TCP 接続で SIP を実行している場合は [**Terminate on BYE**] 設定を無効にする必要があります。

[**Terminate on BYE**] 設定を無効にするには、[Custom] ボックスをクリックして、[**Enabled**] ボックスをオフにします。

SIP ヘッダの取り扱い

SIP プロファイルのオプション機能の 1 つにヘッダの挿入があり、SIP 要求にヘッダを挿入するかどうかを指定できるようになっています。具体的には、Via ヘッダ、Secure Via ヘッダ、Record-Route ヘッダの

挿入を有効化 / 無効化することが可能です。設定済みの SIP プロファイルをバーチャルサーバに割り当てた場合、そのプロファイルで指定されたヘッダが、BIG-IP によってプールまたはプールメンバに送信される SIP 要求に挿入されます。

SIP メッセージへの Via ヘッダの挿入

SIP メッセージの発信元を示す **Via** ヘッダを SIP 要求に挿入するように、BIG-IP を設定することができます。この経路情報を使って、SIP 応答はセッションイニシエータのロケーションを特定します。

デフォルトでは、[**Insert Via Header**] 設定は無効になっています。この設定を有効にするには [Custom] ボックスをクリックします。

ユーザ値の指定

[**Insert Via Header**] 設定を有効にしたら、次に [**User Via**] 値を指定します。[**User Via**] 値とは、先頭の **Via** ヘッダの値として SIP 要求に挿入する文字列であり、SIP プロトコルとあわせて、仮想アドレスとポートを指定するためのものです。[**User Via**] 文字列は、たとえば **SIP/2.0/UDP 10.10.10.10:5060** のようになります。

SIP メッセージへの Secure Via ヘッダの挿入

（TCP 経由で）SSL/TLS を使ってサーバノードでセキュアチャンネルを作成する場合、SIP 要求に Secure Via ヘッダを挿入するように BIG-IP を設定することができます。[**Secure Via Header**] 設定は、SIP メッセージの発信元を示すものです。

デフォルトでは、[**Secure Via Header**] 設定は無効になっています。この設定を有効にするには [Custom] ボックスをクリックします。

SIP メッセージへの Record-Route ヘッダの挿入

また、SIP 要求に Record-Route ヘッダを挿入するように BIG-IP を設定することもできます。[**Insert Record-Route Header**] 設定は、後続の SIP 要求のネクストホップを指定するものです。

デフォルトでは、[**Insert Record-Route Header**] 設定は無効になっています。この設定を有効にするには [Custom] ボックスをクリックします。

SIP パーシステンスのカスタマイズ

BIG-IP での SIP セッションのパーシステンスの扱い方をカスタマイズすることができます。このカスタマイズは、「SIP パーシステンスプロファイル」と呼ばれる特定のパーシステンスプロファイルを作成することで行いますが、SIP パーシステンスプロファイルを使う 場合は、必ず SIP サービスプロファイルを作成• 使用しなければならないため、注意が必要です（「*SIP プロファイル設定の概要*」(6-40 ページ）を参照）。そのためには、両方のプロファイルを作成し 、それらを同じバーチャルサーバに割り 当てます。SIP パーシステンスプロファイルの詳細については、「*SIP パーシステンス*」(7-17 ページ）を参照してください。

RTSP プロファイルの設定

BIG-IP には、Real Time Streaming Protocol（RTSP）トラフィックの管理に使用するためのプロファイルタイプが含まれます。***Real Time Streaming Protocol（RTSP）***はストリーミングメディアプレゼンテーション用のプロトコルであり、クライアントシステムは RTSP を使用することで、リモートのストリーミングメディアサーバを制御し、サーバ上のファイルへの時間ベースのアクセスを可能にします。

BIG-IP の RTSP プロファイルは、以下の機能をサポートしています。

- UDP 経由のストリーミングメディアの設定。この場合、制御接続を使って必要なポートを解放し、BIG-IP 経由でデータを伝送できるようにします。
- 制御接続経由のインタリーブデータ（基本的には TCP 経由のストリーミングメディア）
- RTSP ポート（554）を経由した、RTSP over HTTP の Real Networks トンネリング

バーチャルサーバに RTSP プロファイルを割り当てる場合、TCP プロファイルもあわせて割り当てる必要があります。TCP プロファイルを割り当てずにいると、自動で割り当てられるようになっています。

表 6.20 では、RTSP プロファイルで指定可能な設定を、それぞれの概要とデフォルト値と共に一覧にしています。

汎用プロパティ	説明	デフォルト値
Name	プロファイルのユーザ定義名を指定します。プロファイル名の指定は必須です。	デフォルト値なし
Parent Profile	カスタムプロファイルの派生元とするプロファイルを指定します。	**rtsp**
Idle Timeout	接続が削除対象として適格になる前に、UDP 接続がアイドル状態である秒数を指定します。	**300**
Maximum Header Size	接続が切断されるまでに BIG-IP で対応可能な、RTSP 要求もしくは応答のヘッダの最大サイズをバイト単位で指定します。	**4096**
Maximum Queued Data	接続が利用不可能であると判断され、その後切断されるまでに、BIG-IP がバッファに格納する最大データ量を指定します。	**32768**
Unicast Redirect	ユニキャストストリームを使用していて、この設定を有効にすると、ストリームデータの宛先アドレスとポートをクライアントで指定することができます。ただし、セキュリティ上の理由から、要求の送信元が宛先アドレスとして使用されます。	無効（チェックされていない）
Multicast Redirect	マルチキャストストリームを使用していて、この設定を有効にすると、クライアントがストリームデータ用に異なる宛先を指定することが許可されます。	Disabled（チェックなし）

***表 6.20** RTSP プロファイルの設定*

汎用プロパティ	説明	デフォルト値
Session Reconnect	この設定が有効になっていて、切断されていた制御接続が再開されると、BIG-IP は再開した制御接続を正しいサーバにパーシストします。一般的なクライアントではこの動作はサポートされません。	Disabled（チェックなし）
Real HTTP Persistence	この設定を有効にすると、Real Networks トンネル処理された RTSP over HTTP が、RTSP ポート経由で自動的にパーシストされます。デフォルト値は **Enabled** です。この設定を無効にした場合、ユーザは iRule を使ってデフォルト動作をオーバーライドすることができます。	**Enabled**（チェックあり）
Check Source	チェックされている（有効になっている）場合、BIG-IP はメッセージの発信元を調べて、そのメッセージがクライアントとサーバのどちらから送信されたかを見極めます。	**Enabled**（チェックあり）
Proxy	RTSP プロファイルが RTSP プロキシ設定と関連付けられているかどうかを指定します。指定できる値は、[**None**]、[**External**]、[**Internal**] です。	**None**
Proxy Header	SETUP 要求に挿入するヘッダ名を指定します。名前は必ず **X-** 文字列で始まる必要があります。通常、このヘッダの値はクライアント IP アドレスに関する情報で構成され、他の RTSP プロファイルによって読み出されます。なお、このヘッダは、サーバに要求が送信されて処理が行われる前に削除されるようになっています。	デフォルト値なし
RTP Port	Microsoft® Media Services サーバが使用するポート番号を指定します。通常 Microsoft Media Services では、固定の Realtime Transport Protocol（RTP）ポート番号が使われますが、この設定により、固定 RTP ポート番号の代わりに使用するポート番号を指定することができます。	**0**
RTCP Port	**RTP Port** 値と対になる RTCP ポートを指定します。通常 Microsoft Media Services では、固定の Real-Time Control Protocol（RTCP）ポート番号が使われますが、この設定により、固定 RTCP ポート番号の代わりに使用するポート番号を指定することができます。	**0**

***表 6.20** RTSP プロファイルの設定（続き）*

RTSP プロファイルの設定では、RTSP クライアントとメディアサーバのほか、アカウンティングと認証のタスクを管理する RTSP プロキシが含まれるのが一般的となっています。このような設定では、多くの場合、サーバからのストリーミングメディアが RTSP プロキシサーバを迂回して直接クライアントに渡されることになります。

この設定を実装するには、2 台のバーチャルサーバ（外部ネットワークとの間で送受信されるトラフィックの処理用と、内部ネットワークとの間で送受信されるトラフィックの処理用）を作成し、BIG-IP を設定します。各バーチャルサーバに、別々の RTSP プロファイルを割り当てます。

この設定では、RTSP プロファイルは以下のように動作します。

- 外部バーチャルサーバの RTSP プロファイルは、内部バーチャルサーバの RTSP プロファイルにクライアント IP アドレスの情報を渡します。

- 内部バーチャルサーバの RTSP プロファイルは、要求からクライアント IP アドレスの情報を抽出して、メディアサーバの応答を処理し、BIG-IP の指定されたポートを解放します。これらのポートが解放されることで、サーバからクライアントへのデータ送信の際に、ストリーミングメディアが RTSP プロキシサーバを迂回できるようになります。

このクライアント IP アドレス情報は、RTSP プロファイルで指定した [**Proxy Header**] 設定に保存されます。

iSession プロファイルの設定

BIG-IP には、WAN により分離された 2 台の BIG-IP 間に最適化トンネルを作成する ***iSession*** プロファイルタイプが含まれます。具体的には、iSession プロファイルを有効にすると BIG-IP は以下のように動作します。

- キューに入れられたデータをホールドする。
- 入力ワイヤ速度に基づき、特定のタイプと量のデータを圧縮する。

表 6.21 では、iSession プロファイルで指定可能な設定を、それぞれの概要とデフォルト値と共に一覧にしています。

汎用プロパティ	説明	デフォルト値
Name	プロファイルのユーザ定義名を指定します。プロファイル名の指定は必須です。	デフォルト値なし
Parent Profile	カスタムプロファイルの派生元とするプロファイルを指定します。	**isession**
Mode	このプロファイルを使ってWAN経由のトラフィックを最適化するように指定します。	Enabled
Compression	対称圧縮に使用する方式を指定します。 **off**: 圧縮を行いません。 **deflate**: deflate データ圧縮アルゴリズムを使用します。 **lzo**: Lempel-Ziv-Oberhumer（lzo）データ圧縮アルゴリズムを使用します。 **bzip2**: bzip2 データ圧縮アルゴリズムを使用します。 **adaptive**: 現在のトラフィックにもっとも適した圧縮アルゴリズムを使用します。	deflate
Port Transparency	クライアントが指定した宛先ポートがWANの向こう側でも保持されるように指定します。	**Enabled**
Reuse Connection	ローカルとリモートの 2 つの WAN Optimization Module 間の接続を保存・再利用するように指定します。	**Enabled**
Target Virtual	終了した iSession トラフィック用に、クライアント側の BIG-IP がサーバ側の BIG-IP 上のターゲットバーチャルサーバを選択するのに使用する照合基準を指定します。 **none**: BIG-IP は、終了した iSession トラフィックを直接サーバに送信します。 **host match no isession**: iSession プロファイルを持たないホストバーチャルサーバのみを照合します。 **host match all**: すべてのホストバーチャルサーバから、もっとも近いものを選択します。 **match all**: すべてのバーチャルサーバから、もっとも近いものを選択します。	**none**
Endpoint Pool	2 つの BIG-IP 間のトンネルのエンドポイントとして機能するプールを指定します。この設定は、ペアトンネリングに使用されます。指定できる値は、**None**、および定義済みのプールごとのエントリとなっています。	**None**

***表 6.21** iSession プロファイルの設定*

iSessions プロファイルを使用した Local Traffic Manager のペアトンネリング設定の詳しい実装手順については、『***BIG-IP® Local Traffic Manager: Implementations***』を参照してください。

7

セッションパーシステンスの有効化

- セッションパーシステンスの概要
- パーシステンスの種類とそれらのプロファイル

セッションパーシステンスの概要

BIG-IP® のローカルトラフィック管理機能システムを使用すると、セッションパーシステンスを設定できます。***セッションパーシステンス***を設定すると、BIG-IP はクライアントリクエストに対応した特定のプールメンバなどのセッションデータを追跡して格納します。セッションデータを追跡して格納する主な理由は、セッションの存続中、または後続のセッション中に、クライアントリクエストが必ず同じプールメンバに送信されるようにするためです。

さらに、セッションパーシステンスでは、ユーザ設定やユーザ名、パスワードなど、その他の種類の情報を追跡し、格納できます。

BIG-IP は数種類のセッションパーシステンスを備え、それぞれがセッションデータに関する特定タイプのストレージ要件に対応するように設計されています。実装するパーシステンスの種類は、ショッピングカートや航空券の予約における項目など、クライアント固有の情報をどこに、どのように格納するかによって異なります。

たとえば航空券の予約情報は、すべてのサーバがアクセスできるバックエンドデータベース内に格納すること場合もあれば、クライアントが最初に接続した特定サーバ上や、クライアントのマシンの Cookie 内に格納することもあります。パーシステンスを有効にした場合、戻ってきたクライアントはロードバランシングをバイパスし、代わりに直前に接続したサーバに接続して保存した情報にアクセスします。

BIG-IP は指定された期間のセッションデータを保持します。

セッションパーシステンスを設定する主な手段は、パーシステンスプロファイルを設定して、これをバーチャルサーバに割り当てるというものです。バーチャルサーバを通過するすべてのトラフィックではなく、特定タイプのトラフィックのパーシステンスのみを有効にする場合は、iRule を記述します。

パーシステンスプロファイルの設定

パーシステンスプロファイルは、ユーザがプロファイルをバーチャルサーバに割り当てると自動的にパーシステンスを有効にする設定済みのオブジェクトです。パーシステンスプロファイルを使用すると、パーシステンスの種類を実装するプログラムを記述する必要がありません。

BIG-IP が提供する各種パーシステンスはそれぞれ対応するデフォルトのパーシステンスプロファイルを含みます。これらのパーシステンスプロファイルは、それぞれそのタイプのパーシステンスに対する BIG-IP の動作を定義する設定と設定値を含みます。デフォルトのプロファイルを使用することも、デフォルトに基づいたカスタムプロファイルを作成することもできます。

詳細については、本ガイドの次の章を参照してください。

- パーシステンスプロファイルを設定するには、「*パーシステンスの種類とそれらのプロファイル*」（7-4 ページ）を参照してください。
- プロファイルの概要を知るには、第 5 章「*プロファイルについて*」を参照してください。

iRules を介したセッションパーシステンスの有効化

バーチャルサーバを通過するすべてのセッションに対してパーシステンスのタイプを有効にするパーシステンスプロファイルを設定する代わりに、iRule を記述して特定の要求に対する（特定の Cookie バージョンのみを含む HTTP トラフィックに対してなど）パーシステンスのタイプを有効にすることができます。

また、iRule を使用して、SSL 終了要求を有効にすることもできます。つまり BIG-IP が復号化と再暗号化を実行して、および SSL 証明書認証を処理して終了するように要求できます。このタイプの iRule では、HTTP ヘッダの挿入 iRule コマンドを使用して SSL セッション ID を HTTP リクエストにヘッダとして挿入できます。

この章の後半では、パーシステンスプロファイルを使用したパーシステンスの有効化を中心に説明します。iRule を作成して、パーシステンスを有効にする詳細については、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** と、第 17 章「*iRule の記述*」を参照してください。

セッションパーシステンスを有効にした OneConnect プロファイルの使い方

セッションパーシステンスを設定すると、BIG-IP はクライアントリクエストに対応したプールメンバなどのセッションデータを追跡して格納します。バーチャルサーバに対してパーシステンスプロファイルを設定することで、そのセッションの存続中、または後続のセッションで、クライアントリクエストが必ず同じプールメンバに送信されるようになります。

HTTP リクエストの **Request-URI** ヘッダにセッションデータが保存されるようになっています。しかし、Cookie とユニバーサルパーシステンスでは、BIG-IP がこのヘッダ内のセッションデータを無視し、想定外のノードに要求を送信する場合があります。たとえば、クライアントがインターネットプロキシデバイス経由でバーチャルサーバに要求を送信した場合などに、このような問題が発生します。これを防止するには、OneConnect プロファイルを作成して、OneConnect プロファイルとパーシステンスプロファイルの両方をバーチャルサーバに割り当てます。

続く 2 つのセクションでは、OneConnect プロファイルがセッションパーシステンスに与える影響について解説しています。

OneConnect プロファイルを使用しない HTTP 解析

バーチャルサーバが OneConnect プロファイルを参照しない場合、BIG-IP は TCP 接続ごとにロードバランシングを実行します。TCP 接続の負荷が分散されると、その接続に含まれるすべての要求が同じ 1 つのプールメンバに送信されます。

たとえば、バーチャルサーバが OneConnect プロファイルを参照せずに、クライアントリクエストがプール **A** のノード **A** に最初に送信される場合、BIG-IP はノード **A** に対して Cookie を挿入します。

続いて、同じ TCP 接続で、BIG-IP がプール **B** のノード **B** に対する Cookie を含む要求を次に受信した際、BIG-IP はその Cookie 情報を無視して、ノード B ではなくノード **A** に要求を送信します。

OneConnect プロファイルを使用した HTTP 解析

OneConnect プロファイルを使用することでこの問題を解消することができます。バーチャルサーバが OneConnect プロファイルを参照する場合、BIG-IP は TCP 接続内の要求ごとにロードバランシングを実行することができます。つまり、HTTP クライアントが単一の接続で複数の要求を送信すると、HTTP リクエストがそれぞれ個別に処理されることになります。必要に応じて、BIG-IP は HTTP リクエストを異なる宛先サーバに送信します。

たとえば、バーチャルサーバが OneConnect プロファイルを参照して、クライアントリクエストがプール **A** のノード **A** に最初に送信される場合、BIG-IP はノード **A** に対して Cookie を挿入します。続いて、同じ TCP 接続で、BIG-IP がプール **B** のノード **B** に対する Cookie を含む要求を次に受信した際、BIG-IP はその Cookie 情報を使って、ノード **B** に正しく要求を送信します。

トラブルシューティングのヒント

BIG-IP が **Request-URI** ヘッダのパーシステンス情報を無視し、間違ったノードに要求を送信した場合の問題を緩和するには、以下のことを行います。

- OneConnect プロファイルをバーチャルサーバに関連付ける。
- HTTP プロファイルの [**OneConnect Transformations**] 設定が有効になっていることを確認する。

OneConnect プロファイルと HTTP プロファイルの設定については、『***Configuration Guide for BIG-IP® Local Traffic Management***』を参照してください。

パーシステンスの種類とそれらのプロファイル

パーシステンスのプロファイル設定を構成し、BIG-IP 上にセッションのパーシステンスを設定できます。これらの設定はプロファイルの作成時に行うか、またはプロファイルの設定を変更することでプロファイルの作成後に行えます。プロファイルの設定に関する手順については、第 5 章「*プロファイルについて*」を参照してください。

パーシステンスの種類

パーシステンスプロファイルを使用して有効にできるパーシステンスの種類は次のとおりです。

- **Cookie パーシステンス**
  Cookie パーシステンスでは、クライアントのコンピュータに格納された HTTP Cookie を使用して、クライアントが Web サイトで以前にアクセスした同じサーバに再接続できるようにします。
- **宛先アドレスのアフィニティパーシステンス**
  スティッキーパーシステンスとも呼ばれる宛先アドレスのアフィニティパーシステンスは、TCP プロトコルと UDP プロトコルをサポートし、パケットの宛先 IP アドレスのみに基づいて同じサーバにセッション要求を送信します。
- **ハッシュパーシステンス**
  ハッシュパーシステンスでは、既存の iRule に基づいたパーシステンスハッシュを作成できます。
- **Microsoft® Remote Desktop Protocol パーシステンス**
  Microsoft® Remote Desktop Protocol（MSRDP）パーシステンスでは、Microsoft® Remote Desktop Protocol（RDP）サービスを実行するクライアントとサーバ間でのセッションを追跡します。
- **SIP パーシステンス**
  SIP パーシステンスは、UDP、SCTP、TCP を介して送信された Session Initiation Protocol（SIP）メッセージを受信するサーバに使用される種類のパーシステンスです。
- **送信元アドレスのアフィニティパーシステンス**
  シンプルパーシステンスとしても知られる送信元アドレスのアフィニティパーシステンスは、TCP プロトコルと UDP プロトコルをサポートし、パケットの送信元 IP アドレスのみに基づいて同じサーバにセッション要求を送信します。
- **SSL パーシステンス**
  SSL パーシステンスは、SSL セッション ID を使用して、終了していない SSL セッションを追跡するタイプのパーシステンスです。終了した SSL セッションでパーシステンスを有効にする詳細については、第 9 章「*SSL トラフィックの管理*」および第 17 章「*iRule の記述*」と、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。

- **ユニバーサルパーシステンス**
  ユニバーサルパーシステンスでは、パケットで永続するものを定義する数式を記述できます。iRules™ の場合と同じ数式の構文を使用して記述される数式では、連続したバイトをセッション識別子として定義します。

セッションパーシステンスの基準について

実装するパーシステンスの種類に関係なく、BIG-IP が指定されたクライアントから同じプールメンバにすべての要求を送信する際に使用する基準を指定できます。これらの基準は、クライアント接続を制御するバーチャルサーバに基づきます。これらの基準を指定するには、**Match Across Services**、**Match Across Virtual Servers**、および **Match Across Pools** プロファイル設定を使用してください。これらの設定を理解することは、パーシステンスプロファイルの設定に役立ちます。

Match Across Services 設定の指定

Match Across Services プロファイル設定を有効にすると、BIG-IP は接続を制御するバーチャルサーバが最初のパーシステンス接続を制御するバーチャルサーバと同じ仮想アドレスを持つ場合にだけ、パーシステンスの時間制限内に同じクライアントから受信したすべてのパーシステンス接続要求を同じノードに送信しようとします。クライアントから異なる仮想アドレスを有する他のバーチャルサーバに送信される接続要求、またはパーシステンスを使用しない接続要求は、プールに定義されているロードバランシング方法に従ってロードバランスされます。

たとえば、パーシステンスが有効化されているバーチャルサーバ **v1:http** が **http_pool**（ノード **n1:http** と **n2:http** を含む）を参照し、パーシステンスが有効化されているバーチャルサーバ **v1:ssl** がプール **ssl_pool**（ノード **n1:ssl** と **n2:ssl** を含む）を参照するバーチャルサーバマッピングを設定するとします。

クライアントが **v1:http** への最初の接続を確立すると、プール **http_pool** に割り当てられているロードバランシングアルゴリズムによって **n1:http** がノードとして選択されます。この場合、クライアントが **v1:ssl** に接続すると、BIG-IP はロードバランシング方法ではなく、最初の接続で確立されたパーシステンスセッションを使用して接続要求を受信するプールメンバを決定します。BIG-IP は 3 番目の接続要求を **n1:ssl** に送信します。このノードはパーシステンスセッションを共有するクライアントの最初の接続を現在制御する **n1:http** ノードと同じノードを使用します。

この後、同じクライアントが別の仮想アドレス（たとえば、**v2:ssl**）でバーチャルサーバに接続すると、BIG-IP は、ロードバランシング法を使って接続要求を受信するノードを判断し、新しいパーシステンスセッションの追跡を開始します。BIG-IP が新しいパーシステンスセッションを開始するのは、要求されたバーチャルサーバが、最初のパーシステンス接続要求（**v1**）をホスティングするバーチャルサーバとは異なる仮想アドレス（**v2**）を使用するためです。

◆ 重要

Match Across Services 設定を有効にするには、同じ仮想アドレスを使用しているバーチャルサーバ、および SSL パーシステンスを使用しているバーチャルサーバがバーチャルサーバマッピング内に同じノードアドレスを含んでいる必要があります。

◆ 注

Cookie プロファイルの場合、この設定は Cookie Hash 方式だけに適用されます。

Match Across Virtual Servers 設定の指定

クライアントによって開始された個々の接続をどのバーチャルサーバが制御していても、同じクライアントから要求されたすべてのセッションに対するパーシステンスを維持するように BIG-IP を設定できます。**Match Across Virtual Servers** 設定を有効にすると、BIG-IP はパーシステンスの制限時間内で、同じクライアントから受信したすべてのパーシステンス接続要求を同じノードに送信することを試みます。パーシステンスを利用しないクライアントからの接続要求は、現在選択されているロードバランシング方法に従ってロードバランスされます。

◆ 注

Cookie プロファイルの場合、この設定は Cookie Hash 方式だけに適用されます。

◆ 警告

設定を有効にするには、TCP または SSL パーシステンスでプールを使用しているバーチャルサーバがバーチャルサーバマッピング内で同じメンバアドレスを含んでいる必要があります。

Match Across Pools 設定の指定

Match Across Pools 設定を有効にすると、BIG-IP は、所定のパーシステンスレコードを含むプールを使用できます。デフォルトでは無効に設定されています（解除）。

◆ 警告

この設定を有効にすると、バーチャルサーバが指定したものではないプールにクライアントトラフィックが送信される可能性があります。

Cookie プロファイルの場合、この設定は **Cookie Hash** 方式だけに適用されます。

Cookie パーシステンス

BIG-IP では、HTTP Cookie パーシステンスを使用するよう に設定できます。***Cookie パーシステンス***では、クライアント のコンピュータに格納された HTTP Cookie を使用し、クライアント が Web サイト で以前にアクセスしたのと 同じプールメンバに再接続できるよう にします。

Cookie パーシステンスには次の 4 つの方法があります。

- HTTP Cookie Insert 方式
- HTTP Cookie Rewrite 方式
- HTTP Cookie Passive 方式
- Cookie Hash 方式

使用する方法によって、Cookieがクライアントに戻された際にBIG-IPがこれを処理する方法が変わります。

◆注

*F5 では、HTTP リクエストのロードバランシングが正しく行われるように、Cookie プロファイルに加えて OneConnect プロファイルを設定することを推奨しています。詳細については、「**セッションパーシステンスを有効にした OneConnect プロファイルの使い方**」(7-2 ページ）を参照してください。*

Cookie プロファイル設定について

Cookie パーシステンスを実装するには、デフォルトの **Cookie** プロファイルを使用するか、カスタムプロファイルを作成します。表 7.1 は、Cookie プロファイルを構成する設定項目と値を示します。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Persistence Type	パーシステンスの種類を指定します。この設定は必須です。	**Cookie**
Cookie Method	BIG-IP が使用する Cookie 処理の種類を指定します。詳細については、次の項の「*Cookie Method 設定の指定*」を参照してください。	**HTTP Cookie Insert**
Cookie Name	BIG-IP が検索または挿入する Cookie 名を指定します。	この値はプール名に基づいて自動的に生成されます。
Expiration	Cookie の有効期限を設定します。HTTP Cookie Insert および HTTP Cookie Rewrite 方式だけに適用されます。デフォルト（チェックあり）を使用すると、BIG-IP では、セッション Cookie で指定された有効期限が使用されます。	Enabled（チェックあり）
Hash Offset	Cookie パーシステンスの場合、この設定は Cookie Hash 方式だけに適用されます。	**0**

***表 7.1** Cookie パーシステンスプロファイルの設定*

設定	説明	デフォルト値
Hash Length	Cookie パーシステンスの場合、この設定は Cookie Hash 方式だけに適用されます。	**0**
Timeout	これらの設定は、**Cookie Hash** 方式だけに適用されます。パーシステンスエントリの待機時間を秒単位で指定します。タイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**180**
Mirror Persistence	有効な場合（チェックあり）、アクティブユニットが待機モードになったときに、BIG-IP がパーシステンスレコードをピアにミラーリングすることを指定します。Cookie プロファイルの場合、この設定は **Cookie Hash** 方式だけに適用されます。	Disabled（解除）
Match Across Services	同じ仮想 IP アドレスに送信されるクライアント IP アドレスからのすべてのパーシステンス接続が、同じノードに送信されるように指定します。Cookie プロファイルの場合、この設定は **Cookie Hash** 方式だけに適用されます。詳細については、「*Match Across Services 設定の指定*」（7-5 ページ）を参照してください。	Disabled（解除）
Match Across Virtual Servers	同じクライアント IP アドレスからのすべてのパーシステンス接続が同じノードに送信されるように指定します。Cookie プロファイルの場合、この設定は **Cookie Hash** 方式だけに適用されます。詳細については、「*Match Across Virtual Servers 設定の指定*」（7-6 ページ）を参照してください。	Disabled（解除）
Match Across Pools	BIG-IP がこのパーシステンスエントリを含むプールを使用できるように指定します。Cookie プロファイルの場合、この設定は **Cookie Hash** 方式だけに適用されます。詳細については、「*Match Across Pools 設定の指定*」（7-6 ページ）を参照してください。	Disabled（解除）
Override Connection Limit	チェックされている（有効になっている）場合、プールメンバのコネクションリミットがオーバーライドされるように指定して、クライアントをパーシストさせることができます。仮想接続ごとの制限は厳格に維持され、オーバーライドされることはありません。	Disabled（解除）

***表 7.1** Cookie パーシステンスプロファイルの設定（続き）*

Cookie Method 設定の指定

Cookieプロファイル内で**Cookie Method**設定を4つの値のうちの1つに設定できます。

HTTP Cookie Insert 方式

プロファイル内で **HTTP Cookie Insert** 方式を指定した場合、クライアントの接続先のサーバに関する情報が、サーバからの HTTP 応答のヘッダに Cookie として挿入されます。Cookie には **BIGipServer <pool_name>** という名前が付けられ、接続を処理するサーバのアドレスとポートを含みます。Cookie の有効期限は BIG-IP に設定されたタイムアウトに基づいて設定されます。**HTTP Cookie Insert** は、**Cookie Method** 設定のデフォルト値です。

プロファイル設定 **Mirror Persistence**、**Match Across Services**、**Match Across Virtual Servers**、および **Match Across Pools** は、**HTTP Cookie Insert** 方式には適用されません。これらの設定は、**Cookie Hash** 方式だけに適用されます。

◆ヒント

この種のプロファイルは、Performance（HTTP）タイプのバーチャルサーバに割り当てることができます。

HTTP Cookie Rewrite 方式

HTTP Cookie Rewrite 方式を指定すると、BIG-IP はサーバからクライアントに送信された **BIGipCookie** という名前の **Set-Cookie** ヘッダをインターセプトし、その Cookie の名前と値を書き換えます。新しい Cookie には **BIGipServer<pool_name>** という名前が付けられ、接続を処理するサーバのアドレスとポートを含みます。

◆重要

*できる限り、**HTTP Cookie Passive** 方式ではなくこの方法を使用することをお勧めします。*

HTTP Cookie Rewrite 方式では、サーバによって作成される Cookie を設定する必要があります。**HTTP Cookie Rewrite** 方式を成功させるには、BIG-IP が書き換えを行うため、Web サーバからの空白の Cookie が必要です。Apache バリアントでは、次のエントリを **httpd.conf** ファイルに追加することによって Cookie を各 Web ページのヘッダに追加できます。

```
Header add Set-Cookie BIGipCookie=00000000000000000000000...
```

（Cookie は合計で 120 個のゼロを含む必要があります。）

◆注

下位互換性のために、空白の Cookie が含むゼロの数を 75 個にすることもできます。ただし、このサイズの Cookie では、iRules とパーシステンスを一緒に使用することはできません。

プロファイル設定 **Mirror Persistence**、**Match Across Services**、**Match Across Virtual Servers**、および **Match Across Pools** は、**HTTP Cookie Rewrite** 方式には適用されません。これらの設定は、**Cookie Hash** 方式だけに適用されます。

HTTP Cookie Passive 方式

HTTP Cookie Passive 方式を指定した場合、サーバからの応答内で **Set-Cookie** ヘッダへの空白の挿入や検索は実行されません。この方式では Cookie は設定されません。この方式では、サーバは正しいサーバ情報とタイムアウトで書式化された Cookie を提供します。

◆重要

*できる限り、**HTTP Cookie Passive** 方式ではなく HTTP Cookie Rewrite 方式を使用することをお勧めします。*

HTTP Cookie Passive 方式を成功させるには、Cookie 内に適切なサーバ情報を含んだ Web サーバからの Cookie が必要です。設定ユーティリティを使用する場合は、エンコードが自動的に追加される Cookie 文字列のテンプレートを生成し、このテンプレートを編集して実際の Cookie を作成してください。

たとえば次の文字列は、エンコードが自動的に追加される生成済み Cookie テンプレートで、この場合 **[pool name]** はサーバを含むプールの名前、**336260299** はエンコードされたサーバアドレス、および **20480** はエンコードされたポートです。

```
Set-Cookie:BIGipServer[poolname]=336268299.20480.0000; expires=Sat, 01-Jan-2002 00:00:00
    GMT; path=/
```

このテンプレートから Cookie を作成するには、実際のプール名と有効期限の日時を入力します。

または、次のアドレス（a.b.c.d）の数式を使用してエンコードを実行することもできます。

```
d*(256^3) + c*(256^2) + b*256 +a
```

ポートをエンコードするには、ポートを格納して元に戻す 2 バイトを使用します。したがって、ポート **80** は 80 * 256 + 0 = 20480 になります。ポート **1433**（5 * 256 + 153）は 153 * 256 + 5 = 39173 になります。

Apache バリアントでは、次のエントリを **httpd.conf** ファイルに追加することによって Cookie を各 Web ページのヘッダに追加できます。

```
Header add Set-Cookie: "BIGipServer my_pool=184658624.20480.000; expires=Sat, 19-Aug-2002
    19:35:45 GMT; path=/"
```

プロファイル設定 **Mirror Persistence**、**Match Across Services**、**Match Across Virtual Servers**、および **Match Across Pools** は、**HTTP Cookie Passive** 方式には適用されません。これらの設定は、**Cookie Hash** 方式だけに適用されます。

Cookie Hash 方式

Cookie Hash 方式を指定した場合、Cookie 値は継続的に特定のノードにマッピングされます。クライアントがサイトに戻ると、BIG-IP は、Cookie 情報を使用してクライアントを所定のノードに戻します。この方式では、Web サーバが Cookie を生成する必要があります。HTTP Cookie Insert方式を使用した場合のようにCookieが自動的に作成されることはありません。

宛先アドレスのアフィニティパーシステンス

宛先アドレスのアフィニティパーシステンスを使用すると、サーバの配列を最適化できます。*宛先アドレスのアフィニティパーシステンス*はスティッキーパーシステンスとしても知られ、特定の宛先 IP アドレスについての要求を、要求を送信したクライアントに関係なく同じサーバに送信します。

このタイプのパーシステンスは、キャッシングサーバをロード バランシングする場合に最も有益です。キャッシングサーバはWeb要求をインターセプトし、使用可能であればキャッシュされている Web ページを返します。これらのサーバでのキャッシュの有効性を改善するには、

同様の要求を同じサーバに繰り返し送信する必要があります。宛先アドレスのアフィニティパーシステンスタイプを使用すると、配列内の各サーバ上の Web ページではなく 1 つのサーバ上の Web ページをキャッシュできます。これによって、他のサーバはキャッシュ内の Web ページを複製する必要がなくなり、メモリの無駄を省くことができます。

宛先アドレスのアフィニティプロファイル設定について

宛先アドレスのアフィニティパーシステンスを実装するには、デフォルトの **dest_addr** プロファイルを使用するか、カスタムプロファイルを作成します。表 7.2 は、宛先アドレスアフィニティプロファイルを構成する設定項目と値を示します。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Persistence Type	パーシステンスプロファイルの種類を指定します。この設定は必須です。	**Destination Address Affinity**
Mirror Persistence	有効な場合（チェックあり）、アクティブユニットが待機モードになったときに、BIG-IP がパーシステンスレコードをピアにミラーリングすることを指定します。	Disabled（解除）
Match Across Services	同じ仮想 IP アドレスに送信されるクライアント IP アドレスからのすべてのパーシステンス接続が、同じノードに送信されるように指定します。	Disabled（解除）
Match Across Virtual Servers	同じクライアント IPアドレスからのすべてのパーシステンス接続が同じノードに送信されるように指定します。	Disabled（解除）
Match Across Pools	BIG-IP がこのパーシステンスエントリを含むプールを使用できるように指定します。	Disabled（解除）
Mask	既存のパーシステンスエントリと照合する前に、BIG-IP が使用するマスクを指定します。	**255.255.255.255**
Timeout	パーシステンスエントリの持続時間を秒単位で指定します。タイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。 指定できる値は次のとおりです。 **Specify**: パーシステンスエントリが期限切れになるまでの秒数を指定します。 **Indefinite**: パーシステンスエントリに期限切れがないことを指定します。	**180**
Override Connection Limit	チェックされている（有効になっている）場合、プールメンバのコネクションリミットがオーバーライドされるように指定して、クライアントをパーシストさせることができます。仮想接続ごとの制限は厳格に維持され、オーバーライドされることはありません。	Disabled（解除）

***表 7.2** 宛先アドレスのアフィニティパーシステンスプロファイルの設定*

ハッシュパーシステンス

ハッシュパーシステンスでは、**persist uie** iRule コマンドを使用する既存の iRule に基づきパーシステンスハッシュを作成することができます。ハッシュパーシステンスはユニバーサルパーシステンスと同じように使用されますが、ハッシュパーシステンスでは返されるパーシステンスキーがデータそのものではなくデータのハッシュ値となります。

図 7.1 は、ハッシュパーシステンスを実装する iRule の例です。

```
rule my_persist_irule {
   when HTTP_REQUEST { persist uie [HTTP::header myheader] }
}
```

***図 7.1** ハッシュパーシステンス用の iRule の例*

ハッシュパーシステンス用の iRule 実行後は、**bigpipe persist show all** コマンドを入力して、データそのものではなくデータのハッシュ値が使用されていることを確認することができます。

たとえば、データ文字列 **data1** と **data2** を指定する iRule がある場合、**bigpipe persist show all** コマンドを実行すると、これらの文字列のハッシュ値（**2356372769** と **1996459178**）がパーシステンス値として表示されます。図 7.2 はこのコマンドの出力です。

```
# b persist show all
PERSISTENT CONNECTIONS --
Mode: hash Value: 2356372769
Virtual: 10.1.1.41:http Node: 10.10.10.190:http Age: 2sec
Mode: hash Value: 1996459178
Virtual: 10.1.1.41:http Node: 10.10.10.180:http Age: 2sec
```

***図 7.2** パーシステンスにハッシュ値が使用されている出力結果*

ハッシュパーシステンスを使用しており、BIG-IP が接続のパーシステンステーブルにエントリを見つけられず、フォールバックパーシステンスによりプールメンバを選択できない場合、指定したロードバランシング方式ではなく、ハッシュ値を使用して、プールメンバが選択されます。

前述の例に基づいて、パーシステンステーブルにハッシュ値 **2356372769** のエントリが含まれておらず、プール中のアクティブノード数が同じである場合、パーシステンスにこのハッシュ値を取るセッションが常に、（ノードがアクティブであれば）ノード **10.10.10.190** にパーシステンスされます。

Hash プロファイル設定について

ハッシュパーシステンスを実装するには、デフォルトの**ハッシュ**プロファイルを使用するか、カスタムのプロファイルを作成します。表 7.3 は、Hash プロファイルを構成する設定項目と値を示します。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Persistence Type	パーシステンスプロファイルの種類を指定します。この設定は必須です。	**Hash**
Match Across Services	同じ仮想 IP アドレスに送信されるクライアント IP アドレスからのすべてのパーシステンス接続が、同じノードに送信されるように指定します。	Disabled（解除）
Match Across Virtual Servers	同じクライアント IP アドレスからのすべてのパーシステンス接続が同じノードに送信されるように指定します。	Disabled（解除）
Match Across Pools	BIG-IP がこのパーシステンスエントリを含むプールを使用できるように指定します。	Disabled（解除）
Hash Algorithm	ハッシュパーシステンスのロードバランシングに使用するアルゴリズムを指定します。ハッシュ結果がこのアルゴリズムの入力値となります。指定できる値は次のとおりです。 • **Default**: プールメンバのインデックスを使用して、アルゴリズムの入力値となるハッシュ結果を取得します。 • **CARP**: Cache Array Routing Protocol（CARP）を使用して、アルゴリズムの入力値となるハッシュ結果を取得します。	**デフォルト**
Hash Offset	パケット内の開始オフセットを指定します。ハッシュパーシステンスのロードバランシング実行時には、ここからハッシュが開始されることになります。デフォルト値 **0** はオフセットがないことを示します。	**0**
Hash Length	パケット内のデータの長さをバイト単位で指定します。ハッシュパーシステンスのロードバランシング実行時には、これを使ってハッシュ値が算出されます。	**0**
Hash Start Pattern	ハッシュパターンの開始位置を記述する文字列式（Tcl regex）を指定します。ハッシュパーシステンスのロードバランシング実行時には、この開始位置が使用されます。	デフォルト値なし
Hash End Pattern	ハッシュパターンの終了位置を記述する文字列式（Tcl regex）を指定します。ハッシュパーシステンスのロードバランシング実行時には、この終了位置が使用されます。	デフォルト値なし
Hash Buffer Limit	ハッシュパターンの位置を検索する際の最大データ量を指定します。	**0**
Hash More	現在のハッシュ処理終了後に別のハッシュ処理を実行するように指定します。	無効（チェックされていない）
iRule	パーシステンスエントリを決定するために実行する iRule を指定します。	**None**
Timeout	パーシステンスエントリの待機時間を秒単位で指定します。タイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。 指定できる値は次のとおりです。 **Specify**: パーシステンスエントリが期限切れになるまでの秒数を指定します。 **Indefinite**: パーシステンスエントリに期限切れがないことを指定します。	**180**

***表 7.3** Hash パーシステンスプロファイルの設定*

設定	説明	デフォルト値
Override Connection Limit	チェックされている（有効になっている）場合、プールメンバのコネクションリミットがオーバーライドされるように指定して、クライアントをパーシストさせることができます。仮想接続ごとの制限は厳格に維持され、オーバーライドされることはありません。	Disabled（解除）

表 7.3 Hash パーシステンスプロファイルの設定

Microsoft Remote Desktop Protocol パーシステンス

MSRDP パーシステンスでは、Microsoft® Remote Desktop Protocol（RDP）サービスを実行している Windows® クライアントとサーバとの間でトラフィックのロードバランシングとパーシステンスセッションの維持を効率的に実行できます。MSRDP パーシステンス機能を有効にする際に推奨されるシナリオは、Windows Server 2003 または Windows Server 2008 を実行するメンバで構成されるロードバランシングプールを作成するというものであり、ここではすべてのメンバが Windows クラスタに所属し、Windows セッションディレクトリに参加することになります。

MSRDP パーシステンスのメリット

通常、Microsoft Terminal Services を実行する Windows サーバでは、セッションブローカ（Windows Server 2003 では「***Terminal Services Session Directory***」、Windows Server 2008 では「***TS Session Broker***」と呼ばれています）を利用して、ユーザセッションが特定のサーバに割り当てられるようにすることができます。クライアントが誤ったターミナルサーバに対して接続要求を開始すると、そのサーバがクライアントを正しいサーバにリダイレクトします。

しかし、BIG-IP を使用している場合は、ロードバランシングプール内の個々のサーバではなく、BIG-IP のバーチャルサーバにクライアントをリダイレクトする必要があります。このようなリダイレクトが行われるようにするには、MSRDP プロファイルを設定します。MSRDP プロファイルと共に、セッションブローカがパーシステンス記録の保持用に提供するトークンが使用されます。セッションブローカトークンが存在しないセッションが開始された場合、BIG-IP はそのプールで設定されているロードバランシング方式に基づき、どのようなロードバランシングを行うかを決定します。

つまり、MSRDP パーシステンスとセッションブローカを併用すると、セッションブローカを単独で使用する場合よりも、ロードバランシングの選択肢の範囲が広がり、拡張性と柔軟性が向上することになります。

サーバプラットフォームの問題

デフォルトでは、BIG-IP で MSRDP パーシステンスを有効にすると、プール内の各サーバでセッションブローカが設定されている限り、ロードバランシング方式の設定に従い接続の負荷が分散されます。Terminal Services Session DirectoryはWindows Server 2003で、TS Session Broker は Windows Server 2008 で、それぞれ限定して提供されている機能です。したがって、プール内のサーバが Windows Server 2003 か

Windows Server 2008 のいずれかでなければ、MSRDP パーシステンスをデフォルトモードで使用することはできません。また、クライアントシステムでは、あらゆる Windows Server 2003 または Windows Server 2008 システムに付属するリモートデスクトップクライアントソフトウェアが実行されている必要があります。

ただし、MSRDP パーシステンスを有効にする必要はあるが、サーバプラットフォームが旧バージョンの Windows（Session Directory または TS Session Broker が使用不可）を実行しているという場合は、MSRDP パーシステンスを非デフォルトモードにすることができます。この場合、BIG-IP はクライアントが指定したユーザ名を経由して同じ Windows サーバにクライアントを接続します。MSRDP パーシステンスを非デフォルトモードで有効にする（つまり、サーバでセッションブローカが利用できない）と、ロードバランシングとリダイレクションの機能が制限されるため、デフォルトモードで使用する方がより望ましいといえます。

セッションブローカを有効にした MSRDP パーシステンスの設定

MSRDP パーシステンスをデフォルトモードで有効にするには、ロードバランシングプール内の各 Windows サーバでセッションブローカを設定する必要があります。セッションブローカの設定に加え、その他の Windows 設定タスクをこれらのサーバで実行します。ただし、Windows サーバを設定する前に、ロードバランシングプールを作成したり、Windows サーバをそのプールのメンバとして指定するといったタスクを実行することによって、BIG-IP を設定する必要があります。

次の 2 つの項では、RDP を使用した Windows クライアント / サーバの設定に関して、MSRDP パーシステンスをデフォルト モード で有効にするために必要な BIG-IP のシステム設定タスクについて説明します。

セッションブローカを無効にした MSRDP パーシステンスの設定

サーバにセッションブローカがない場合、サーバは他のサーバとセッションを共有できないため、サーバへの接続が切断された場合にリダイレクションを実行できません。セッション共有の代わりに、Windows クライアントはユーザ名の形式でデータを BIG-IP に提供します。BIG-IP はこれによってそのクライアントを同じサーバに継続的に接続することができます。MSRDP パーシステンスがこのように動作するように有効化することは、非デフォルトモードです。

MSRDP プロファイル設定について

MSRDP パーシステンスを実装するには、デフォルトの **msrdp** プロファイルを使用するか、カスタムのプロファイルを作成します。表 7.4 は、プロファイルの MSRDP タイプを構成する設定項目と値を示します。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Persistence Type	パーシステンスプロファイルの種類を指定します。この設定は必須です。	**Microsoft Remote Desktop**
Mirror Persistence	有効な場合（チェックあり）、アクティブユニットが待機モードになったときに、BIG-IP がパーシステンスレコードをピアにミラーリングすることを指定します。	Disabled（解除）
Match Across Services	同じ仮想 IP アドレスに送信されるクライアント IP アドレスからのすべてのパーシステンス接続が、同じノードに送信されるように指定します。	Disabled（解除）
Match Across Virtual Servers	同じクライアント IPアドレスからのすべてのパーシステンス接続が同じノードに送信されるように指定します。	Disabled（解除）
Match Across Pools	BIG-IP がこのパーシステンスエントリを含むプールを使用できるように指定します。	Disabled（解除）
Timeout	パーシステンスエントリの持続時間を秒単位で指定します。タイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。 指定できる値は次のとおりです。 **Specify**: パーシステンスエントリが期限切れになるまでの秒数を指定します。 **Indefinite**: パーシステンスエントリに期限切れがないことを指定します。	**300**
Has Session Directory	サーバが Session Directory を実行するかどうかを指定します。	**Enabled**（チェックあり）
Override Connection Limit	チェックされている（有効になっている）場合、プールメンバのコネクションリミットがオーバーライドされるように指定して、クライアントをパーシストさせることができます。仮想接続ごとの制限は厳格に維持され、オーバーライドされることはありません。	Disabled（解除）

表 7.4 MSRDP パーシステンスプロファイルの設定

SIP パーシステンス

Session Initiation Protocol（SIP）は、複数の参加者で構成されるセッションを管理するアプリケーションレイヤプロトコルであるため、リアルタイムのメッセージ、音声、データ、および映像が有効化されます。セッションは電話やインスタントメッセージによる単純な双方向の会話の場合もあれば、音声、データ、ビデオで構成される複雑でコラボレーティブなマルチメディアカンファレンスコールの場合もあります。SIP を使用した場合、アプリケーションは SCTP、TCP、または UDP を介してメッセージを交換することで、相互に通信を行います。

SIP パーシステンスはサーバプールで利用可能なパーシステンスの一種です。SIP パーシステンスは、UDP を介して送信された SIP メッセージを受信するプロキシサーバに対して設定することができます。現在 BIG-IP では、UDP、TCP、SCTP 経由で送信された SIP メッセージのパーシステンスをサポートしています。

SIP パーシステンスのプロファイル設定について

SIP パーシステンスを実装するには、デフォルトプロファイルである **Persistence Type** か **SIP** を利用するか、、もしくはカスタムプロファイルを作成します。表 7.5 に、SIP パーシステンスプロファイルを構成する設定項目と値をまとめています。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Persistence Type	パーシステンスプロファイルの種類を指定します。この設定は必須です。	**SIP**
Parent Profile	カスタムプロファイルの派生元とするプロファイルを指定します。	**sip_info**
Match Across Services	同じ仮想 IP アドレスに送信されるクライアント IP アドレスからのすべてのパーシステンス接続が、同じ ノードに送信されるように指定します。	Disabled（解除）
Match Across Virtual Servers	同じクライアント IP アドレスからのすべてのパーシステンス接続が同じ ノードに送信されるように指定します。 すべてのバーチャルサーバに対してパーシステンスを設定すると、クライアント によって開始された個々の接続がどのバーチャルサーバで制御されているかに関係なく、トラフィック管理システムは同じクライアント が要求するすべての接続のパーシステンスを維持すること になり ます。	Disabled（解除）
Match Across Pools	BIG-IP がこのパーシステンスエントリを含むプールを使用できるように指定します。 すべてのプールに対してパーシステンスを設定すると、クライアントによって開始された個々の接続がどのプールで制御されているかに関係なく、トラフィック管理システムは同じクライアントが要求するすべての接続のパーシステンスを維持することになります。	Disabled（解除）

表 7.5 SIP パーシステンスプロファイルの設定

設定	説明	デフォルト値
SIP Info	SIP セッションをパーシストさせる SIP ヘッダフィールドを指定します。以下のヘッダフィールドから選択してください。 Call-ID: セッションが Call-ID でパーシストするように指定します。 SIP-ETag: セッションがSIP-ETagでパーシストするように指定します。 To: セッションが SIP セッションの宛先でパーシストするように指定します。 From: セッションが SIP セッションの発信元でパーシストするように指定します。 **Subject**: セッションが SIP セッションのサブジェクトでパーシストするように指定します。 **Specify**: セッションが SIP ヘッダにおけるユーザ指定の文字列でパーシストするように指定します。	**From**
Timeout	パーシステンスエントリの持続時間を秒単位で指定します。タイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。 指定できる値は次のとおりです。 **Specify**: パーシステンスエントリが期限切れになるまでの秒数を指定します。 **Indefinite**: パーシステンスエントリに期限切れがないことを指定します。	**180**
Override Connection Limit	チェックされている（有効になっている）場合、プールメンバのコネクションリミットがオーバーライドされるように指定して、クライアントをパーシストさせることができます。仮想接続ごとの制限は厳格に維持され、オーバーライドされることはありません。	Disabled（解除）

***表 7.5** SIP パーシステンスプロファイルの設定（続き）*

タイムアウト値を指定することによって、BIG-IP は異なる種類の SIP ファイナルメッセージが 1 つあるだけで各受信パケットをテストする必要がなくなり、旧 SIP パーシステンスエントリに関連付けられたリソースを開放できます。デフォルトのタイムアウト値は **180** 秒です。タイムアウト値を変更する場合、デフォルト以上の値を指定することをお勧めします。

◆ 重要

*UDP トラフィックを処理するバーチャルサーバの場合、SIP プロファイルの **Timeout** 設定値（秒）が最低でも UDP プロファイルの **Idle Timeout** 設定値と同じ値になっていることを必ずチェックするようにします。これによって、SIP トラフィックのパーシステンスが正しく行われていることを確認できます。*

送信元アドレスのアフィニティパーシステンス

*送信元アドレスのアフィニティパーシステンス*はシンプルパーシステンスとしても知られ、送信元IPアドレスのみに基づいてセッションを追跡します。クライアントが送信元アドレスのアフィニティパーシステンスをサポートするバーチャルサーバへの接続を要求すると、BIG-IPはクライアントが以前に接続していないかどうかを確認し、接続していた場合は、クライアントを同じプールメンバに返します。

送信元アドレスのアフィニティパーシステンスと SSL パーシステンスを一緒に使用することもあります。SSL セッション ID がタイムアウトしている、または戻されたクライアントがセッション ID を提供しない場合、BIG-IPにクライアントのIPアドレスに基づいて元のプールメンバにクライアントをダイレクトさせる必要があります。クライアントの送信元アドレスのアフィニティパーシステンスレコードがタイムアウトになっていない限り、BIG-IP はクライアントを適切なプールメンバに戻すことができます。

パーシステンスの設定はすべてのプロトコルに適用されます。パーシステンスのタイマーが **0** より大きい値に設定されている場合、パーシステンスは**オン**になります。パーシステンスのタイマーが **0** に設定されている場合、パーシステンスは**オフ**になります。

パーシステンスマスク機能は、送信元アドレスのアフィニティパーシステンスを実装するバーチャルサーバについてのみ機能します。パーシステンスマスクを追加することによって、送信元 IP アドレスの範囲を特定し、プールに接続する際に 1 つの送信元アドレスのアフィニティパーシステンス接続として一緒に管理できます。

送信元アドレスのアフィニティパーシステンスプロファイル設定について

送信元アドレスのアフィニティパーシステンスを実装するには、デフォルトの **source_addr** プロファイルを使用するか、カスタムプロファイルを作成します。表 7.6 は、送信元アドレスアフィニティプロファイルを構成する設定項目と値を示します。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Persistence Type	パーシステンスプロファイルの種類を指定します。この設定は必須です。	**Source Address Affinity**
Match Across Services	同じ仮想IPアドレスに送信されるクライアントIPアドレスからのすべてのパーシステンス接続が、同じノードに送信されるように指定します。	Disabled（解除）
Match Across Virtual Servers	同じクライアント IPアドレスからのすべてのパーシステンス接続が同じノードに送信されるように指定します。	Disabled（解除）
Match Across Pools	BIG-IP がこのパーシステンスエントリを含むプールを使用できるように指定します。	Disabled（解除）

***表 7.6** 送信元アドレスのアフィニティパーシステンスプロファイルの設定*

設定	説明	デフォルト値
Timeout	パーシステンスエントリの持続時間を秒単位で指定します。タイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。 指定できる値は次のとおりです。 **Specify**: パーシステンスエントリが期限切れになるまでの秒数を指定します。 **Indefinite**: パーシステンスエントリに期限切れがないことを指定します。	**180**
Mask	既存のパーシステンスエントリと照合する前に、BIG-IP が使用するマスクを指定します。	**None**
Map Proxies	プロキシのマッピングを有効または無効にします。	**Enabled**（チェックあり）
Override Connection Limit	チェックされている（有効になっている）場合、プールメンバのコネクションリミットがオーバーライドされるように指定して、クライアントをパーシストさせることができます。仮想接続ごとの制限は厳格に維持され、オーバーライドされることはありません。	Disabled（解除）

***表 7.6** 送信元アドレスのアフィニティパーシステンスプロファイルの設定*

SSL パーシステンス

SSL パーシステンスは SSL セッション ID を使用して SSL セッションを追跡するタイプのパーシステンスで、個々のプールのプロパティです。クライアントが、通常はインターネットのサービスプロバイダが割り当てるような変換された IP アドレスまたはダイナミック IP アドレスを持つ場合、SSL パーシステンスの使用は特に重要です。クライアントの IP アドレスが変更された場合でも、BIG-IP はセッションをセッション ID に基づいてパーシステンスがあるとして認識します。

SSL パーシステンスと送信元アドレスのアフィニティパーシステンスを一緒に使用することもあります。SSL セッション ID がタイムアウトしている、または戻されたクライアントがセッション ID を提供しない場合、BIG-IP にクライアントの IP アドレスに基づいて元のノードにクライアントをダイレクトさせる必要があります。クライアントのシンプルパーシステンスレコードがタイムアウトになっていない限り、BIG-IP はクライアントを適切なノードに戻すことができます。

◆注

SSL パーシステンスは、クライアントリクエストまたはサーバ応答に対する SSL 証明書ベースの認証を実行しないシステムのみに有効です。Client SSL または Server SSL プロフィルを使用して証明書ベースの認証を設定している場合は、SSL パーシステンスプロフィルを設定しないでください。設定する代わりに、SSL セッションパーシステンスを実行する iRule を作成します。

SSL パーシステンスプロファイル

SSL パーシステンスを実装するには、デフォルトの **ssl** プロファイルを使用するか、カスタムのプロファイルを作成します。表 7.7 は、SSL パーシステンスプロファイルを構成する設定項目と値を示します。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Persistence Type	パーシステンスプロファイルの種類を指定します。この設定は必須です。	**SSL**
Match Across Services	同じ仮想 IP アドレスに送信されるクライアント IP アドレスからのすべてのパーシステンス接続が、同じノードに送信されるように指定します。	Disabled（解除）
Match Across Virtual Servers	同じクライアント IPアドレスからのすべてのパーシステンス接続が同じノードに送信されるように指定します。	Disabled（解除）
Match Across Pools	BIG-IP がこのパーシステンスエントリを含むプールを使用できるように指定します。	Disabled（解除）
Timeout	パーシステンスエントリがタイムアウトになるまでの秒数を指定します。つまり、この設定によって、SSL セッション ID のタイムアウト値が設定されます。この値によって、指定された SSL セッション ID がどのくらいの間削除されずに BIG-IP に格納されるかが決まります。タイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**
Override Connection Limit	チェックされている（有効になっている）場合、プールメンバのコネクションリミットがオーバーライドされるように指定して、クライアントをパーシストさせることができます。仮想接続ごとの制限は厳格に維持され、オーバーライドされることはありません。	Disabled（解除）

表 7.7 SSL パーシステンスプロファイルの設定

ユニバーサルパーシステンス

BIG-IP の Universal Inspection Engine（UIE）には、BIG-IP の iRules 内に指定できる一連の機能が含まれ、これによってトラフィックをより細かい方法でダイレクトします。これらの iRule 機能を使用すると、コンテンツデータに基づいてトラフィックをダイレクトする、またはトラフィックをプールの特定のメンバにダイレクトする数式を記述できます。

ユニバーサルパーシステンスではこの iRules 機能をさらに一歩進め、コンテンツデータ、またはプールの特定メンバへの接続に基づき、iRule **persist uie** コマンドを使ってセッションのパーシステンスを実装できるようになっています。ユニバーサルパーシステンスでは、セッション識別子として使用する一連のバイトを定義することによってこれを実行します。

パーシステンスの iRule 数式を使用するために、ユニバーサルパーシステンスプロファイルには数式を含む iRule の名前を指定する設定が含まれています。

図 7.3 は、ユニバーサルパーシステンスを実装する iRule の例です。

```
rule my_persist_irule {
   when HTTP_REQUEST { persist uie [HTTP::header myheader] }
}
```

図 7.3 ユニバーサルパーシステンス用 iRule の例

データのハッシュ値をパーシステンスキーとして使用するハッシュパーシステンスと違い、ユニバーサルパーシステンスではデータそのものをパーシステンスキーとして使用します。

iRule を作成してユニバーサルプロファイル内で iRule 名を指定したら、iRule とプロファイルの両方をリソースとして適切なバーチャルサーバに割り当てる必要があります。

iRules の詳細については、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** または第 17 章「*iRule の記述*」を参照してください。

◆注

*F5 では、HTTP リクエストのロードバランシングが正しく行われるように、ユニバーサルプロファイルに加えて OneConnect プロファイルを設定することを推奨しています。詳細については、「**セッションパーシステンスを有効にした OneConnect プロファイルの使い方**」（7-2 ページ）を参照してください。*

ユニバーサルパーシステンスプロファイル

ユニバーサルパーシステンスを実装するには、デフォルトの**ユニバーサル**プロファイルを使用するか、カスタムのプロファイルを作成します。表 7.8 は、ユニバーサルパーシステンスプロファイルを構成する設定項目と値を示します。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Persistence Type	パーシステンスの種類を指定します。この設定は必須です。	**Universal**
Match Across Services	同じ仮想 IP アドレスに送信されるクライアント IP アドレスからのすべてのパーシステンス接続が、同じノードに送信されるように指定します。	Disabled（解除）
Match Across Virtual Servers	同じクライアント IPアドレスからのすべてのパーシステンス接続が同じノード に送信されるよう に指定します。	Disabled（解除）
Match Across Pools	BIG-IP がこのパーシステンスエントリを含むプールを使用できるように指定します。	Disabled（解除）
iRule	BIG-IP がパーシステンスエントリを決定するために実行する既存の iRule の名前を指定します。	**None**

表 7.8 ユニバーサルパーシステンスプロファイルの設定

設定	説明	デフォルト値
Timeout	パーシステンスエントリの持続時間を秒単位で指定します。タイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。 指定できる値は次のとおりです。 **Specify**: パーシステンスエントリが期限切れになるまでの秒数を指定します。 **Indefinite**: パーシステンスエントリに期限切れがないことを指定します。	**180**
Override Connection Limit	チェックされている（有効になっている）場合、プールメンバのコネクションリミットがオーバーライドされるように指定して、クライアントをパーシストさせることができます。仮想接続ごとの制限は厳格に維持され、オーバーライドされることはありません。	Disabled（解除）

表 7.8 ユニバーサルパーシステンスプロファイルの設定

8

プロトコルプロファイルの管理

- プロトコルプロファイルの概要
- Fast L4 プロファイルの設定
- Fast HTTP プロファイルの設定
- HTTP Class プロファイルの設定
- TCP プロファイルの設定
- UDP プロファイルの設定
- SCTP プロファイルの設定

プロトコルプロファイルの概要

一部の BIG-IP プロファイルはユーザによる設定が可能で、「プロトコルプロファイル」と呼ばれています。プロトコルプロファイルには、次のタイプがあります。

- Fast L4
- Fast HTTP
- HTTP Class
- TCP
- UDP
- SCTP

BIG-IP では、プロトコルプロファイルのタイプごとに、デフォルト設定を持つ事前設定されたプロファイルを使用できます。ほとんどの場合、これらのデフォルトプロファイルはそのまま使用できます。これらの設定を変更する場合は、プロファイルの作成時、またはプロファイルの作成後にプロファイルの設定を変更することで、プロトコルプロファイルの設定を行います。

この章の以降の項では、Fast L4、Fast HTTP、HTTP Class、TCP、UDP、SCTP プロファイルに含まれるトラフィック管理設定を示します。その他のタイプのプロファイルの設定については、次を参照してください。

- HTTP、FTP、RTSP、SIP プロファイルについては、第 6 章「*アプリケーション層トラフィックの管理*」を参照してください。
- セッションパーシステンスに使用するプロファイルについては、第 7 章 *セッションパーシステンスの有効化*」を参照してください。
- クライアントおよびサーバ SSL プロファイルについては、第 9 章「*SSL トラフィックの管理*」を参照してください。
- リモート認証サーバ経由のトラフィック認証に使用されるプロファイルについては、第 10 章「*アプリケーショントラフィックの認証*」を参照してください。
- OneConnect、NTLM、Statistics、Stream プロファイルについては、第 11 章「*その他のプロファイルの使用*」を参照してください。

Fast L4 プロファイルの設定

Fast L4 プロファイルの目的は、レイヤ 4 トラフィックを効率よく管理できるようにすることです。バーチャルサーバに Fast L4 プロファイルを割り当てると、BIG-IP 内の Packet Velocity® ASIC（PVA）ハードウェア高速化が、通過するレイヤ 4 トラフィックのすべてまたは一部を処理できます。レイヤ 4 の処理を PVA ハードウェア高速化へ負荷を緩和することで、BIG-IP は基本ルーティング機能（レイヤ 4）およびアプリケーション切り替え（レイヤ 7）のパフォーマンスとスループットを向上することができます。

Fast L4 プロファイルは、パフォーマンス（ レイヤ 4)、転送（ レイヤ 2)、および転送 IP)というバーチャルサーバのタイプで使用できます。

Fast L4 プロファイル設定について

デフォルトの **fastl4** プロファイルをそのまま使用することも、カスタム Fast L4 プロファイルを作成することもできます。標準的なニーズについては、Fast L4 プロファイル設定の大部分のデフォルト値で対応できます。変更が必要となる設定は、[**Reset on Timeout**]、および [**Idle Timeout**] です。

◆注

*既存の Fast L4 プロファイルに行った変更は、**Idle Timeout** 値が終了した後、または接続が終了した後で、接続だけで有効になります。*

表 8.1 に、Fast L4 プロファイルの設定を列挙して説明します。

設定	説明	デフォルト値
Name	この設定は、プロファイルの一意の名前を指定します。	デフォルト値なし
Parent Profile	この設定は、親プロファイルとして使用するプロファイルを指定します。新しいプロファイルは、指定された親プロファイルから非カスタム設定および値をすべて継承します。	**fastL4**
Reset on Timeout	この設定が有効になっており、TCP 接続がアイドル接続のタイムアウト値を超えた場合、BIG-IP によって接続が削除され、リセットが送信されます。	**Enabled**
Reassemble IP Fragments	この設定が有効になっている場合、BIG-IP によって IP フラグメントが再構成されます。	**Disabled**
Idle Timeout	この設定は、接続が削除対象として適格になる前に、接続がアイドル状態である秒数を指定します。アイドルタイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**

***表 8.1** Fast L4 プロファイルの設定*

設定	説明	デフォルト値
TCP Handshake Timeout	**Specify**: TCP ハンドシェイクの許容可能な期間を指定します。これは、クライアント SYN とクライアント ACK の間の最大アイドル時間です。TCP ハンドシェイクがタイムアウトより長くなると、この接続は自動的に終了します。 **Disabled**: BIG-IP で TCP ハンドシェイクにタイムアウトを適用しないことを指定します。 **Indefinite**: TCP ハンドシェイクの許容可能な期間が無制限であることを指定します。	**5**
Max Segment Size Override	最大セグメントサイズ（MSS）の 1460 をオーバーライドします。指定できる値は次のとおりです。 **Disabled**: 最大セグメントサイズを 1460 のままにすることを指定します。 **Specify**: 値を指定して、最大セグメントサイズ（1460）のオーバーライドを許可します。値 **0** を指定すると、デフォルト値のままになります（**Disabled**）。	**Disabled**
PVA Acceleration	この設定は、BIG-IP で使用したい最大高速化モードを指定します。バーチャルサーバの構成に基づいて、BIG-IP は、このモードで、トラフィックの高速化を実行することも、実行しないこともあります。（詳細については、「*PVA ハードウェア高速化の設定*」（8-5 ページ）を参照してください。）指定できる値は、**[Full]**、**[Assisted]**、または **[None]** です。この設定の詳細については、次の表を参照してください。	**Full**
IP ToS to Client	この設定は、クライアントへの UDP パケット送信時に BIG-IP によって IP パケットに割り当てられる Type of Service（ToS）レベルを指定します。	**Pass Through**
IP ToS to Server	この設定は、サーバへの UDP パケット送信時に BIG-IP によって IP パケットに割り当てられる Type of Service（ToS）レベルを指定します。	**Pass Through**
Link QoS to Client	この設定は、クライアントへの UDP パケット送信時に BIG-IP によって IP パケットに割り当てられる Quality of Service（QoS）レベルを指定します。	**Pass Through**
Link QoS to Server	この設定は、サーバへの UDP パケット送信時に BIG-IP によって IP パケットに割り当てられる QoS レベルを指定します。	**Pass Through**
TCP Timestamp Mode	BIG-IP が TCP タイムスタンプで行う処理を指定します。指定できる値は、**[Preserve]**、**[Strip]**、および **[Rewrite]** です。	**Preserve**
TCP Window Scale Mode	BIG-IP が TCP ウィンドウで行う処理を指定します。指定できる値は、**Preserve** と **Strip** です。	**Preserve**
Generate Internal Sequence Numbers	BIG-IP が RFC 1948 に従って SYN パケットの独自のシーケンス番号を生成することを可能にします。	Disabled
Strip Sack OK	BIG-IP が開始 SYN におけるサーバへの受け渡しから TCP SackOK オプションをブロックすることを可能にします。	Disabled
RTT from Client	BIG-IP がクライアントへのラウンドトリップタイムを測定するために TCP タイムスタンプオプションを使用するように指定します。	Disabled

***表 8.1** Fast L4 プロファイルの設定（続き）*

設定	説明	デフォルト値
RTT from Server	BIG-IP がサーバへのラウンドトリップタイムを測定するために TCP タイムスタンプオプションを使用するように指定します。	Disabled
Loose Initiation	チェックされている（有効になっている）場合、TCP パケットを受信すると、接続開始に SYN パケットを要求せずに、接続を初期化することを指定します。デフォルトでは無効になっています。**Loose Initiation** 設定を有効にする場合、**Loose Close** 設定も有効にすることをお勧めします。 **重要** : Loose Initiation を有効にすると、迷子パケットが BIG-IP 内を通過することがあります。これは、セキュリティリスクとなり、システムパフォーマンスを低下させる場合があります。	Disabled
Loose Close	チェックされている（有効になっている）場合、BIG-IP がクライアントまたはサーバから最初の FIN パケットを受信すると、Loose Initiation 接続を終了することを指定します。	Disabled
TCP Close Timeout	BIG-IPが該当する接続でCLOSEパケットを受信したときに、接続が削除される前にアイドル状態を取ることができる時間を秒単位で指定します。**TCP Close Timeout** 値は、**Idle Timeout** 値よりも小さくする必要があります。また、**TCP Close Timeout** 値は、**Loose Initiation** または **Loose Close** 設定が有効になっている場合にのみ許可されます。	**5**
Hardware SYN Cookie Protection	システム上に PVA10 がある場合に、ハードウェア SYN Cookie 保護を有効化 / 無効化します。この機能は特定のハードウェアプラットフォームでのみ使用可能です。	Disabled
Software SYN Cookie Protection	システム上に PVA10 がない場合に、ソフトウェア SYN Cookie 保護を有効化 / 無効化します。	Disabled

***表 8.1** Fast L4 プロファイルの設定（続き）*

PVA ハードウェア高速化の設定

Fast L4 プロファイルを実装すると、BIG-IP は、レイヤ 4 トラフィックで最も効率が良い PVA ハードウェア高速化モードを自動的に選択します。可能なモードは、**[Full]**、**[Assisted]**、または **[None]** です。

BIG-IP が選択するハードウェア高速化モードは、以下の要素によって異なります。

- Fast L4 プロファイルの設定
  BIG-IP が選択するモードは、Fast L4 プロファイルの設定内容に影響を受けます。
- **バーチャルサーバの設定**
  BIG-IP が選択するモードは、バーチャルサーバに割り当てた所定の機能（プール、SNAT プール、および iRules など）によって影響を受けます。
- **対応するノードに割り当てられたモニタ**
  PVA のフル高速化の場合、該当するノードにモニタを割り当てる必要があります。
- **PVA Acceleration 設定の値**
  Fast L4 プロファイルの **PVA Acceleration** 設定により、バーチャルサーバを通過するレイヤ 4 トラフィックに対して、許可したい最大ハードウェア高速化が定義されます。したがって、各値の設定は以下のようになります。
  - **Full**: BIG-IP は、バーチャルサーバの設定に基づいて、3 つのモード（**Full**、**Assisted**、または **None**）のいずれにもハードウェア高速化を設定できます。これはデフォルト値です。
  - **Assisted**: BIG-IP は、バーチャルサーバの設定に基づいて、**Assisted** または **None** のいずれかにハードウェア高速化を設定できます。
  - **None**: BIG-IP は、ハードウェア高速化を実行しません。

最大ハードウェア高速化設定を **Full** 以下に設定する理由のひとつに、**bigpipe conn show** コマンドで接続を表示する場合があります。このコマンドは、ハードウェア高速化モードが **Assisted** または **None** に設定されている場合に、レイヤ 4 接続だけを表示します。モードが **Full** に設定されている場合、**bigpipe conn show** コマンドでは、レイヤ 4 接続が表示されません。

ハードウェア高速化が自動的に設定された現在のモードに基づいて、BIG-IP は、表 8.2（8-6 ページ）に示したように、レイヤ 4 トラフィックを高速化します。

◆ 重要

*BIG-IP で VLAN グループが設定され、**Transparency Mode** 設定が **[Translucent]** か **[Transparent]** になっている場合、**[PVA Acceleration]** 値は自動的に **[None]** に設定されます。*

ハードウェア高速化モード	結果
Full	ハードウェア高速化がすべてのレイヤ4トラフィックを処理します。レイヤ4トラフィックは、BIG-IP ソフトウェア機能を使用して管理され*ません*。この場合、BIG-IP は、クライアント側およびサーバ側のパケットを同じ接続の一部として処理します。 **Full** モードのハードウェア高速化を使用する例としては、Round Robin ロードバランシング方式を使用して、セッションパーシステンスまたは iRules なしで、2 つのサーバ間でレイヤ 4 トラフィックのロードバランシングを行う場合があります。
Assisted	BIG-IP は、すべての SYN パケットのロードバランシングを行い、ハードウェア高速化は、接続の切断を含め、残りのパケットを処理します。 **Assisted**モードのハードウェア高速化を使用する例としては、ダイナミックロードバランシング方式を使用して、またはパケットに含まれる IP アドレスを確認するシンプルな iRule を使用して、レイヤ 4 トラフィックのロードバランシングを行う場合があります。 ***注:*** *BIG-IP が、ハードウェア高速化を* ***Assisted*** *に設定した場合、Fast L4 プロファイルは、SNAT および SNAT プールと、送信元アドレスのアフィニティパーシステンスと互換性を持ちます。*
None	ハードウェア高速化はレイヤ 4 トラフィックを処理しません。代わりとして、BIG-IP アプリケーションがすべてのレイヤ 4 トラフィックを管理します。 **None** モードのハードウェア高速化を使用する例としては、HTTP プロファイルや、遅延バインディングおよび Cookie セッションパーシステンスを実行する iRule を使用してトラフィックのロードバランシングを行う場合があります。

表 8.2 *レイヤ4 トラフィックに対する PVA ハードウェア高速化の効果*

Fast HTTP プロファイルの設定

Fast HTTP プロファイルは、特定タイプの HTTP 接続を加速するために設計された設定ツールです。このプロファイルは、TCP、HTTP、および OneConnect プロファイルの選択した機能を、最適なネットワークパフォーマンスが得られるように最適化された1つのプロファイルに統合します。このプロファイルをバーチャルサーバに関連付けると、バーチャルサーバによってトラフィックがパケットごとに、大幅に高速に処理されるようになります。

以下のような場合、Fast HTTP の使用を検討する必要があります。

- セッションパーシステンス、リモート サーバ認証、SSLトラフィック管理、および TCP 最適化などの機能や、データ圧縮、パイプライニング、および RAM キャッシュなどの HTTP 機能が必要ない場合。
- 送信元 IP アドレスを保持する必要がない場合。
- 宛先サーバに対して開く接続数を減らしたい場合。
- 宛先サーバが、接続パーシステンス、すなわち、**Keep-Alive** ヘッダを含む HTTP/1.1 または HTTP/1.0 をサポートしている場合。IIS サーバの場合、デフォルトで接続パーシステンスをサポートしています。
- 必要なのは、限定されたレイヤ 4 サポートや限定された HTTP ヘッダ処理などの、基本 iRule だけです。たとえば、iRule イベント **CLIENT_ACCEPTED**、**SERVER_CONNECTED**、および **HTTP_REQUEST** を使用できます。

Fast HTTP プロファイルを使用する大きな利点は、プロファイルが接続パーシステンスをサポートする様態にあります。Fast HTTP プロファイルを使用すると、クライアントリクエストに対して、BIG-IP は、HTTP **Connection** ヘッダを変換または追加して、接続を開いたままにできます。また、このプロファイルを使用することで、BIG-IP は開いているサーバ側接続をプーリングできます。このような接続パーシステンスのサポートにより、接続を開いたり、閉じたりすることで生じるオーバーヘッドの大部分が排除され、宛先サーバの負荷が大幅に軽減されます。HTTP ヘッダ変換の詳細については、第 6 章「*アプリケーション層トラフィックの管理*」を参照してください。サーバ側接続のプーリングの詳細については、第 11 章「*その他のプロファイルの使用*」を参照してください。

◆注

Fast HTTP プロファイルは、その他すべてのプロファイルタイプと非互換です。また、このプロファイルタイプを VLAN グループとともに使用したり、IPv6 アドレスフォーマットで使用することはできません。

Fast HTTP プロファイル設定について

デフォルトの **fasthttp** プロファイルをそのまま使用することも、カスタム Fast HTTP プロファイルを作成することもできます。表 8.3 に、Fast HTTP プロファイルの設定を列挙して説明します。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。	デフォルト値なし
Parent Profile	親プロファイルとして使用するプロファイルを指定します。新しいプロファイルは、指定された親プロファイルから非カスタム設定および値をすべて継承します。	**fasthttp**
Reset on Timeout	チェックされている（有効になっている）場合、接続がタイムアウトになると BIG-IP によって TCP RESET パケットが送信され、接続が削除されます。	**Enabled**（チェックあり）
Idle Timeout	この設定は、トラフィックがないために接続フローが削除対象として適格になる前に、接続がアイドル状態である秒数を指定します。指定できる値は、**[Specify]**、**[Immediate]**、および **[Indefinite]** です。アイドルタイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**
Maximum Segment Size Override	サーバ側接続に対する最大セグメントサイズ（MSS）オーバーライドを指定します。デフォルト設定は **0** で、**1460** の MSS に相当します。**536** ～ **1460** の任意の整数を指定すると、この値をオーバーライドできます。	**0**
Client Close Timeout	BIG-IP がクライアント FIN パケットを受信するか、またはクライアントに FIN パケットを送信する場合に、クライアント接続を閉じるまでの秒数を指定します。この設定は **[Idle Timeout]** 設定をオーバーライドします。指定できる値は、**[Specify]**、**[Immediate]**、および **[Indefinite]** です。詳細については、オンラインヘルプを参照してください。	**5**
Server Close Timeout	BIG-IP がサーバ FIN パケットを受信するか、またはサーバに FIN パケットを送信する場合に、クライアント接続を閉じるまでの秒数を指定します。この設定は **[Idle Timeout]** 設定をオーバーライドします。指定できる値は、**[Specify]**、**[Immediate]**、および **[Indefinite]** です。詳細については、オンラインヘルプを参照してください。	**5**
Unclean Shutdown	接続のクローズ処理方法を指定します。指定できる値は、**[Disabled]**、**[Enabled]**、および **[Fast]** です。詳細については、オンラインヘルプを参照してください。	**Disabled**
Force HTTP 1.0 Response	チェックされている（有効になっている）場合、サーバはクライアントへの応答を HTTP/1.0 フォーマットで送信します。これにより、クライアントのチャンクおよびパイプライニングが効率的に無効にされます。	Disabled（解除）
Maximum Pool Size	ロードバランシングプールで受け付け可能な接続の最大数を指定します。**0** を設定すると、最大数なしを指定します。これは、プールが無制限の数の接続を受け入れられることを意味します。	**2048**
Minimum Pool Size	ロードバランシングプールで受け付け可能な接続の最小数を指定します。**0** を設定した場合、最小数はありません。	**0**

***表 8.3** Fast HTTP プロファイルの設定*

設定	説明	デフォルト値
Ramp-Up Increment	利用可能な接続がすべて使用中である場合に、BIG-IP で追加の接続が利用可能になる増分を指定します。	**4**
Maximum Reuse	現在の接続を再利用できる最大回数を指定します。	**0**
Idle Timeout Override	接続にトラフィックがない場合に、プール内のサーバ側接続が削除対象として適格になるまでの秒数を指定します。この設定は **[Idle Timeout]** 設定をオーバーライドします。指定できる値は、**[Specify]**、**[Disabled]**、および **[Indefinite]** です。詳細については、オンラインヘルプを参照してください。	**Disabled**
Replenish	BIG-IP によってバックエンド接続の定常状態の最大数を維持するかどうかを指定します。この設定を無効にした場合、プールへの接続数が **[Minimum Pool Size]** 設定で指定された値を下回らないかぎり、BIG-IP によってバックエンドへの接続の定常状態の最大数は維持されません。	**Enabled**（チェックあり）
Parse Requests	チェックされている（有効になっている）場合、BIG-IP によって接続ストリームで HTTP データが解析されます。非 HTTP トラフィックに対して Fast HTTP プロファイルを使用している場合は、ダイナミックなサービス拒否（DDoS）攻撃を遮断するために、この設定を無効にする必要があります。	**Enabled**（チェックあり）
Maximum Header Size	ロードバランシングに関する決定を行う前に、BIG-IP によってバッファに入れられるHTTPヘッダデータの最大量を指定します。	**32768**
Maximum Requests	BIG-IP で単一のクライアント側接続に許可する要求の最大数を指定します。指定された制限に達した場合、最終応答には **Connection: close** ヘッダが含まれ、その後に接続のクローズが続きます。デフォルト設定の **0** を指定すると、BIG-IP ではクライアント側接続ごとに無限数の要求が許可されます。	**0**
Insert XForwarded For	コネクションプーリングで使用するために、**XForwarded For:** ヘッダをクライアント IP アドレスとともに HTTP リクエストに挿入するかどうかを指定します。可能な設定は、**[Enabled]** および **[Disabled]** です。詳細については、オンラインヘルプを参照してください。	**Disabled**
Request Header Insert	BIG-IP によって HTTP リクエストにヘッダとして挿入する文字列を指定します。ヘッダがすでに存在する場合、置換は行われません。	デフォルト値なし

***表 8.3** Fast HTTP プロファイルの設定（続き）*

iRules を作成するときに、Fast HTTP プロファイルでサポートされる多数のイベントおよびコマンドを指定できます。これらの iRule イベントおよびコマンドの詳細については、DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** と、第 17 章「*iRule の記述*」を参照してください。

HTTP Class プロファイルの設定

HTTP Class プロファイルは、HTTP トラフィックの分類に使用するための設定ツールです。*トラフィック分類*では、トラフィックヘッダやコンテンツの調査に基づき、トラフィックを宛先に転送します。HTTP Class プロファイルを使用することは、BIG-IP がユーザ指定の基準に基づきトラフィックを分類するための有効な手段です。iRule 機能を利用して同じトラフィック分類関数を実行することもできますが、HTTP Class プロファイルを使った方が手間がかかりません。

ロードバランシングプールか URL のいずれかを宛先として指定することができます。HTTP トラフィックの分類を行うには、リストタイプと照合する文字列を指定する HTTP Class プロファイルを設定します。文字列照合に使用可能なリストタイプは、次のとおりです。

- ホスト名
- URI
- ヘッダ
- Cookie

これらのリストのいずれかと照合可能な文字列は、パターン文字列もしくは正規表現となります。

リストタイプでは、パターン文字列の大文字と小文字が区別されます。たとえば、**www.f5.com** というパターン文字列は **www.F5.com** とは別のものとして扱われます。大文字と小文字の区別を無効にするには、Linux **regexp** コマンドを使用します。

文字列と対応するリストタイプとがマッチすると、BIG-IP はユーザ指定のプールにトラフィックを送信することができます。もしくは、クライアントリクエストをターゲットの HTTP バーチャルサーバから、プールではなく HTTPS バーチャルサーバに転送する HTTP Class プロファイルを作成することも可能です。

HTTP Class プロファイルの設定について

デフォルトの **httpclass** プロファイルをそのまま使用するか、もしくはカスタムの HTTP Class プロファイルを作成することもできます。表 8.4（8-11 ページ）に、HTTP Class プロファイルの設定を一覧にまとめ、説明しています。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。	デフォルト値なし
Parent Profile	親プロファイルとして使用するプロファイルを指定します。新しいプロファイルは、指定された親プロファイルから非カスタム設定および値をすべて継承します。	**httpclass**
Application Security	バーチャルサーバが Application Security Managerアプリケーションにトラフィックを転送するように指定します。この場合、HTTP Class プロファイルは Application Security Manager のアプリケーションセキュリティクラスに相当します。この設定は、BIG-IP で Application Security Manager の使用が許可されている場合にかぎり表示されます。	Disabled（解除）
WebAccelerator	バーチャルサーバが WebAccelerator™ アプリケーションにトラフィックを転送するように指定します。この設定は、BIG-IP で WebAccelerator の使用が許可されている場合にかぎり表示されます。	Disabled（解除）
Hosts	HTTPリクエストのルーティングの基準として使用されるホスト名が、すべてのホストのものであるか、ユーザ指定のホストのものであるかを指定します。[**Match All**] にするとすべてのホストの HTTP リクエストが転送され、[**Match Only**]にするとユーザ指定のホストに基づき HTTP リクエストの転送が行われます。	**Match All**
Host List	HTTPリクエストのルーティングの基準となるホスト名を指定します。さらに、**Entry Type** リストを使用して、各ホスト名をパターン文字列もしくは正規表現として指定する必要があります。この設定は、**Hosts** 値が [**Match Only**] の場合にのみ表示されます。 **注 :** *パターン文字列を使用する場合、リストタイプの大文字と小文字が区別されます 詳細については、「**HTTP Class プロファイルの設定**」(8-10 ページ)を参照してください。*	デフォルト値なし
URI Paths	HTTPリクエストのルーティングの基準として使用される URI が、すべての URI であるか、ユーザ指定の URI であるかを指定します。[**Match All**] にするとすべての URI の HTTP リクエストが転送され、[**Match Only**] にするとユーザの指定の URI に基づき HTTP リクエストの転送が行われます。	**Match All**
URI List	HTTPリクエストのルーティングの基準となるURIパスを指定します。さらに、**Entry Type** リストを使用して、各 URI をパターン文字列もしくは正規表現として指定する必要があります。この設定は、**URI Paths** 値が [**Match Only**] の場合にのみ表示されます。 **注 :** *パターン文字列を使用する場合、リストタイプの大文字と小文字が区別されます 詳細については、「**HTTP Class プロファイルの設定**」(8-10 ページ)を参照してください。*	デフォルト値なし
Headers	HTTPリクエストのルーティングの基準として使用されるヘッダとそれらの値が、すべてのヘッダのものであるか、ユーザ指定のヘッダのものであるかを指定します。[**Match All**] にするとすべてのヘッダを基準に HTTP リクエストが転送され、[**Match Only**] にするとユーザ指定のヘッダに基づき HTTP リクエストの転送が行われます。	**Match All**

表 8.4 HTTP Class プロファイルの設定

設定	説明	デフォルト値
Header List	HTTPリクエストのルーティングの基準として使用されるヘッダとそれらの値を指定します。さらに、**Entry Type** リストを使用して、各ヘッダをパターン文字列もしくは正規表現として指定する必要があります。この設定は、**Headers** 値が [**Match Only**] の場合にのみ表示されます。 **注 :** *パターン文字列を使用する場合、リストタイプの大文字と小文字が区別されます 詳細については、「**HTTP Class プロファイルの設定」(8-10 ページ)** を参照してください。*	デフォルト値なし
Cookies	HTTPリクエストのルーティングの基準として使用されるCookie が、すべての Cookie であるか、ユーザ指定の Cookie であるかを指定します。[**Match All**] にするとすべての Cookie を基準に HTTP リクエストが転送され、[**Match Only**] にするとユーザ指定の Cookie に基づき HTTP リクエストの転送が行われます。	**Match All**
Cookie List	HTTP リクエスト HTTP リクエストのルーティングの基準となる Cookie を指定します。さらに、**Entry Type** リストを使用して、各 Cookie をパターン文字列もしくは正規表現として指定する必要があります。この設定は、**Cookies** 値が [**Match Only**] の場合にのみ表示されます。 **注 :** *パターン文字列を使用する場合、リストタイプの大文字と小文字が区別されます 詳細については、「**HTTP Class プロファイルの設定」(8-10 ページ)** を参照してください。*	デフォルト値なし
Send To	HTTPトラフィックの宛先を指定します。指定できる値は、[**None**]、[**Pool**]、または [**Redirect To**] です。	**None**
Pool	分類されたトラフィックの送信先となるプールの名前を指定します。この設定は、**Send To** 設定が [**Pool**] の場合にのみ表示されます。	**None**
Redirect To Location	トラフィックの送信先となる URI を指定します。この設定は、クライアントリクエストを HTTP バーチャルサーバから、プールではなく HTTPS バーチャルサーバに転送したい場合に使用します。たとえば、ポート **80** でリッスンするように、URL**http://siterequest/** の HTTP バーチャルサーバを作成します。次に、このバーチャルサーバにHTTP Class プロファイルを割り当て、クライアントリクエストを HTTPS バーチャルサーバ **https://siterequest/** にリダイレクトするようにします。 文字列は Tcl 形式で指定することが可能です（**https://[HTTP::host][HTTP::uri]** など）。	デフォルト値なし
Rewrite URI	HTTP リダイレクトをクライアントに送信することなく、サーバに転送される要求 URI をリライトするために使用する TCL 形式を指定します。この設定で、TCL 形式ではなく静的テキストを使用した場合、システムはすべての受信要求用に指定された URI をマップします。また、**Send To** の設定値が [**Redirect To**] の場合はこの設定を使用することはできません。	デフォルト値なし

***表 8.4** HTTP Class プロファイルの設定（続き）*

ほかの BIG-IP モジュールによるトラフィック分類

BIG-IP に Application Security Manager や WebAccelerator システムが搭載されている場合、先にそのモジュールに HTTP トラフィックを送信してから、最終的な宛先への送信を行うように BIG-IP を設定することができます。たとえば、HTTP Class プロファイルを使って、Application Security Manager経由でトラフィックを送信した後に、ロードバランシングプールにそのトラフィックを転送するように、バーチャルサーバに指示することが可能です。

HTTP Class プロファイルは、設定ユーティリティの [Local Traffic] セクションから作成できるほか、Application Security Manager もしくは WebAccelerator システムでも作成することが可能です。なお、Application Security Manager においてトラフィック分類を行う場合、HTTP Class プロファイルは*アプリケーションセキュリティクラス*と呼ばれます。

モジュールで HTTP Class プロファイル（あるいはアプリケーションセキュリティクラス）を作成する場合、そのモジュールはデフォルトで有効になっています。それに対して、設定ユーティリティの [Local Traffic] セクションから HTTP Class プロファイルを作成する場合は、同プロファイルで **Application Security** もしくは **WebAccelerator** 設定を明示的に有効にする必要があります。この設定を明示的に有効にしなければ、関連するバーチャルサーバでそのモジュールが無効になります。

Application Security Manager のアプリケーションセキュリティクラスと WebAccelerator の HTTP Class プロファイルの設定については、以下の資料を参照してください。

- 『***Configuration Guide for BIG-IP® Application Security Management***』
- 『***Configuration Guide for the BIG-IP® WebAccelerator™ System***』

TCP プロファイルの設定

TCP プロファイルは、TCP ネットワークトラフィックを管理する設定ツールです。TCP プロファイル設定の多くは標準 SYSCTL タイプの設定ですが、BIG-IP に固有の設定もあります。

TCP プロファイルは、一部のタイプの別のプロファイルを実装するときに必要となるため、非常に重要です。たとえば、パーシステンスプロファイルやリモート認証プロファイルと合わせて、TCP、HTTP、および OneConnect プロファイルを実装することで、以下のトラフィック管理機能を利用できるようになります。

- サーバの負荷を緩和するコンテンツスプーリング
- サーバ側の接続をプーリングする OneConnect
- ハッシュまたは Cookie Persistence などのレイヤ 7 セッションパーシステンス
- HTTP トラフィックを管理するための iRules
- HTTP RAM キャッシュ
- HTTP データ圧縮
- HTTP パイプライニング
- リモートサーバを使用したアプリケーション認証
- HTTP リダイレクションの書き換え

BIG-IP には、デフォルトの TCP プロファイル（**tcp**）のほか、F5 がユーザ用に作成した 2 つのカスタム TCP プロファイル（**tcp-lan-optimized**、**tcp-wan-optimized**）が含まれます。これらのプロファイルのいずれか 1 つをそのまま実装するか、もしくは必要に応じて設定値を変更することも可能です。

TCP プロファイル設定について

デフォルトの **tcp** プロファイルをそのまま使用することも、カスタム TCP プロファイルを作成することもできます。表 8.5（8-15 ページ）に、TCP プロファイルの設定を列挙して説明します。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。	デフォルト値なし
Parent Profile	親プロファイルとして使用するプロファイルを指定します。新しいプロファイルは、指定された親プロファイルから非カスタム設定および値をすべて継承します。	**tcp**
Reset on Timeout	この設定が有効になっており、TCP 接続がアイドル接続のタイムアウト値を超えた場合、接続が削除され、リセットが送信されます。	**Enabled**（チェックあり）
Time Wait Recycle	SYN パケットが TIME-WAIT 状態で受信された場合に接続をリサイクルします。	**Enabled**（チェックあり）
Delayed ACKs	この設定が有効になっている場合、複数の肯定確認応答（ACK）の合体が許可されます。	**Enabled**（チェックあり）
Proxy Maximum Segment	クライアントとネゴシエーションされたのと同じ最大セグメントをサーバにアドバタイズします。	Disabled（解除）
Proxy Options	クライアントとネゴシエーションされたオプションのみ（タイムスタンプなど）をサーバにアドバタイズします。	Disabled（解除）
Proxy Buffer Low	受信ウィンドウが開かれたプロキシバッファレベルを指定します。	**4096**
Proxy Buffer High	受信ウィンドウが閉じられたプロキシバッファレベルを指定します。	**16384**
Idle Timeout	接続が削除対象として適格になる前に、接続がアイドル状態である秒数を指定します。アイドルタイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**
Time Wait	接続が CLOSED 状態になる前に TIME-WAIT 状態であるミリ秒数を指定します。	**2000**
FIN Wait	接続が終了する前に FIN-WAIT 状態または CLOSING 状態である秒数を指定します。値 **0** は、永久期間（または FIN 状態のメトリックまで）を表します。	**5**
Close Wait	接続が終了する前に LAST-ACK 状態のままになる秒数を指定します。値 **0** は、永久期間（または FIN 状態のメトリックまで）を表します。	**5**
Send Buffer	BIG-IP にバッファサイズ（バイト単位）を送信させます。	**32768**
Receive Window	BIG-IP にウィンドウサイズ（バイト単位）を受信させます。	**32768**
Keep Alive Interval	BIG-IP にプローブ間隔（秒単位）をキープアライブさせます。	**1800**
Maximum SYN Retransmissions	BIG-IP で許可される SYN セグメントの再送信の最大数を指定します。	**3**
Maximum Segment Retransmissions	BIG-IP で許可されるデータセグメントの再送信の最大数を指定します。	**8**
IP ToS	クライアントへのパケット送信時に、BIG-IP によって TCP パケットに割り当てられる Type of Service（ToS）レベルを指定します。	**0**

***表 8.5** TCP プロファイルの設定*

設定	説明	デフォルト値
Link QoS	クライアントへのパケット送信時に、BIG-IP によって TCP パケットに割り当てられる Quality of Service（QoS）レベルを指定します。	**0**
Selective ACKs	この設定がチェックされている（有効になっている）場合、BIG-IP では、システムパフォーマンスを向上させるために、可能な場合は常に選択的 ACK を使用してデータを処理します。この設定を有効にすると、受信したパケットの障害を確認できることから、損失の多いネットワークのパケットフローが円滑化されます。このオプションはネゴシエーションに対応しており、ピアでサポートされていなければ自動で無効になります。 ***注：****F5 ではデフォルト値の使用を推奨しています。*	**Enabled**（チェックあり）
Extended Congestion Notification	この設定がチェックされている（有効になっている）場合、BIG-IP では TCP フラグ **CWR**（Congestion Window Reduction）および **ECE**（ECN-Echo）を使用して、そのピアに輻輳および輻輳対策を通知します。 ***注：****F5 ではデフォルト設定の使用を推奨しています。有効にした場合、この設定が輻輳計算全体の妨げとなる可能性があります。また、セキュリティ問題の発生を許すことになり、****CWR*** *パケットのスプーフィング（なりすまし）によって中間デバイスのパフォーマンスの低下を招くことにもなりかねません。*	Disabled（解除）
Extensions for High Performance (RFC 1323)	この設定がチェックされている（有効になっている）場合、BIG-IP では、TCP に対してタイムスタンプおよびウィンドウスケーリング拡張を使用して（RFC 1323 に準拠）、高速ネットワークパフォーマンスを強化します。これらのオプションは、ラウンドトリップ時間、ならびにピア上の有効なリソースの計算に使用され、輻輳制御と基本的に結びついています。また、通常これらのオプションはネゴシエーションに対応しており、ネットワークデバイスやピアで正しく実装されているかぎり、無効にする必要はありません。	**Enabled**（チェックあり）
Limited Transmit Recovery	この設定がチェックされている（有効になっている）場合、BIG-IP では、高速再送信のために限定送信回復リビジョンを使用して（RFC 3042 に準拠）、損失の多いネットワーク上での接続の回復時間を短縮します。この設定を有効にすると、重複 ACK パケットの最初の受信時に TCP が輻輳ウィンドウを一時的に引き延ばすことができます。また、すばやい再送信を可能にし、小さな輻輳ウィンドウからの復旧も早まります。この設定を有効にした場合、復旧時にかぎり、積極的な転送が行われます。	**Enabled**（チェックあり）
Slow Start	この設定がチェックされている（有効になっている）場合、BIG-IP では、より大きい初期ウィンドウサイズを使用して（RFC 3390 に準拠）、ラウンドトリップタイムを短縮できるようにします。この設定によって、一定期間にピアに送信されるデータの量が増加します。これを有効にすることで、そのリンクでの突然の輻輳や過剰な輻輳の発生を回避することができます。また、輻輳メトリックキャッシュによってピアの履歴データが提供され、Slow Start のジャンプスタートが可能になる場合もあります。 この設定を無効にすると、BIG-IP は輻輳ウィンドウが最大になるように初期化を行い、輻輳が発生するまでできるだけ多くのデータを転送しようとします。そのため、帯域幅に制限のないネットワーク（直接接続されたローカルピアなど）では、最初により多くのデータを転送することが可能です。	**Enabled**（チェックあり）

表 8.5 *TCP プロファイルの設定（続き）*

設定	説明	デフォルト値
Deferred Accept	この設定がチェックされている（有効になっている）場合、BIG-IP では、クライアントからペイロードを受信するまで、接続チェーンコンテキストの割り当てを据え置きます。スリーウェイハンドシェイク DoS 攻撃を処理する場合、この設定を有効にすると役立ちます。	Disabled（解除）
Verified Accept	この設定が有効になっている場合、SYN-ACK を使ってクライアントの SYN に応答する前に、(サーバに実際に SYN を送信することで) サーバが接続を受け入れられる状態にあることを確認します （通常、BIG-IP はクライアントの接続を受け入れた後に、どのサーバと通信を行うかを選択します)。	Disabled（解除）
Bandwidth Delay	この設定がチェックされている（有効になっている）場合、BIG-IP では、スループットおよびラウンドトリップタイムに基づいて、利用可能な帯域幅を超えない範囲で、クライアントへの最適な帯域幅を計算しようとします。	**Enabled**（チェックあり）
Nagle's Algorithm	チェックされている（有効になっている）場合、BIG-IP では Nagle のアルゴリズムを適用して、ネットワーク上の短いセグメントの数を減らします。BIG-IP が最大セグメントサイズ（MSS）よりも小さなパケットを受信すると、ピアが前のセグメントの ACK パケットを送信するまで、それらのパケットはひとまとめに統合されます。これによって、ネットワークで作られるパケットの数が減り、輻輳の緩和につながります。 Telnet などの対話型プロトコルに対してこの設定を有効にした場合、高遅延ネットワークにおいて性能低下を引き起こす可能性があります。	**Enabled**（チェックあり）
Acknowledge on Push	有効になっている場合、非常に小さな送信バッファに書き込んでいる Windows® および MacOS ピアで、大幅なパフォーマンスの改善を指定します。	Disabled（解除）
MD5 Signature	有効になっている場合、RFC2385 TCP-MD5 シグネチャを使用して、中途改ざんから TCP トラフィックを保護することを指定します。	Disabled（解除）
MD5 Signature Passphrase	有効になっている場合、1 ～ 80 文字長の平文パスフレーズを指定します。これは、RFC2385 のなりすまし防止部分を実装するため、共有秘密方式で使用されます。	デフォルト値なし
Congestion Control	BIG-IP が使用する輻輳制御機構を指定します。指定できる値は次のとおりです。 **None** - 輻輳制御アルゴリズムは実装しません。 **High Speed** - より積極的で、損失を重視したアルゴリズム。 **New Reno** - Reno アルゴリズムの改良版。選択的確認応答（SACK）が利用できない場合に、部分的確認応答に対応します。 **Reno** - BSD Reno リリースの実装に基づいた、TCP Fast Recovery アルゴリズムの実装。 **Scalable** - スケーラブルな遅延および損失重視のコンポーネントを Reno アルゴリズムに追加した、TCP アルゴリズムの改良版。	**New Reno**

表 8.5 TCP プロファイルの設定（続き）

設定	説明	デフォルト値
Congestion Metrics Cache	チェックされている（有効になっている）場合、BIG-IP が輻輳メトリックの格納にキャッシュを使用することを指定します。その後は、これらのメトリックは既知となり、すでにキャッシュされてもいるため、前に遭遇したことのあるピアの slow-start 初期ランプが向上します。	**Enabled**（チェックあり）
Appropriate Byte Counting（RFC 3465）	各 ACK が対応するまだ未応答のバイト数の増加に基づいて、輻輳ウィンドウを増加します。 ***注：****F5 ではデフォルト設定の使用を推奨しています。この設定を無効にすると、ACK パケットの損失時に、輻輳ウィンドウが小さいままの時間が延びることになります。*	**Enabled**（チェックあり）
D-SACK（RFC 2883）	重複するセグメントを確認応答するため、**Selective ACKs**（SACK）オプションの使用を指定します。ピアから重複セグメントが送信されなければ、SACK 処理はすべてまとめて無効になります。この設定を有効にした場合、すべての重複セグメントに対して常に SACK を適用するために、より多くの処理が必要になります。	Disabled（解除）
Packet Lost Ignore Rate	100 万パケットあたりの損失数を指定するものであり、この閾数に達したときに輻輳制御が実行されます。指定できる値の範囲は 0 ～ 1,000,00 です。デフォルト値は 0 であり、これは、パケットロスが 1 つでも発生すれば輻輳制御が実行されることを意味しています。Packet Lost Ignore Rate を 10 に設定した場合、TCP 接続のパケットロス数が 10/100 万を超えれば、輻輳制御が発生します。	**0**
Packet Lost Ignore Burst	これを指定することで、たとえ **Packet Lost Ignore Rate** の設定値を上回っていなくても、複数のパケットが失われた場合に輻輳制御を実行させることができます。指定できる値の範囲は **0** ～ **4,294,967,295** です。**0** と指定すると、パケットロスが 1 つでも発生すれば輻輳制御が実行されます。設定値が大きくなればなるほど、輻輳制御の実行の機会が減ることになります。	**0**

表 8.5 *TCP プロファイルの設定（続き）*

TCP プロファイル設定のほとんどについて、通常はデフォルト値がニーズを満たします。ただし、クライアントがバーチャルサーバへのアクセスに使用しているリンクの速度が遅い場合、あるいはサーバ応答時間がクライアントの要求時間を超えた場合、プロファイルのコンテンツスプーリング設定を増やすことができます。

- **Proxy Buffer Low**
- **Proxy Buffer High**
- **Send Buffer**
- **Receive Window**

これらの設定のバイト数を増やすと、該当するデータを受け付ける所定の接続を待機する間に、BIG-IP がバッファできるデータの量が増大します。

◆注

*テスト環境で TCP プロファイルを使用する場合は、**[Slow Start]**、**[Bandwidth Delay]**、および **[Nagle's Algorithm]** 設定を無効にすることでパフォーマンスを向上させることができます。*

tcp-lan-optimized プロファイルの設定について

tcp-lan-optimized プロファイルは TCP タイププロファイルの一種です。BIG-IP でユーザ用に事前作成されたカスタムプロファイルであり、デフォルトの **tcp** プロファイルから派生したものとなっています。

tcp-lan-optimized プロファイルは **tcp** プロファイルの設定とそれらのデフォルト値を継承していますが、中には変更されている設定値もあります。**tcp-lan-optimized** プロファイルを実装することで、カスタムプロファイルを作成しなくても、いくつかの方法によりローカル TCP トラフィックのパフォーマンスを最適化することができます。

tcp-lan-optimized プロファイルをそのまま使用するか、もしくは **tcp-lan-optimized** プロファイルを親プロファイルとして別のカスタムプロファイルを作成することも可能です。

tcp-lan-optimized プロファイルのデフォルト設定値は、表 8.6 の値を除いて、**tcp** プロファイルのものと同じになっています。

設定	説明	デフォルト値
Proxy Buffer Low	受信ウィンドウが開かれたプロキシバッファレベルを指定します。	**98304**
Proxy Buffer High	受信ウィンドウが閉じられたプロキシバッファレベルを指定します。	**131072**
Send Buffer	BIG-IP にバッファサイズ（バイト単位）を送信させます。	**65535**
Receive Window	BIG-IP にウィンドウサイズ（バイト単位）を受信させます。	**65535**
Slow Start	この設定がチェックされている（有効になっている）場合、BIG-IP では、より大きい初期ウィンドウサイズを使用して（RFC 3390 に準拠）、ラウンドトリップタイムを短縮できるようにします。	Disabled（解除）
Bandwidth Delay	この設定がチェックされている（有効になっている）場合、BIG-IP では、スループットおよびラウンドトリップタイムに基づいて、利用可能な帯域幅を超えない範囲で、クライアントへの最適な帯域幅を計算しようとします。	Disabled（解除）
Nagle's Algorithm	チェックされている（有効になっている）場合、BIG-IP では Nagle のアルゴリズムを適用して、ネットワーク上の短いセグメントの数を減らします。デフォルトでは無効に設定されています。テルネットなどの対話型プロトコルに対してこの設定を有効にした場合、高遅延ネットワークにおいて性能低下を引き起こす可能性があります。	Disabled（解除）
Acknowledge on Push	有効になっている場合、非常に小さな送信バッファに書き込んでいる Windows および MacOS ピアで、大幅なパフォーマンスの改善を指定します。	**Enabled**（チェックあり）

表 8.6 tcp プロファイルのものと異なる tcp-lan-optimized プロファイルの設定値

tcp-wan-optimized プロファイルの設定について

tcp-wan-optimized プロファイルは TCP タイププロファイルの一種です。BIG-IP でユーザ用に事前作成されたカスタムプロファイルであり、デフォルトの **tcp** プロファイルから派生したものとなっています。

tcp-wan-optimized プロファイルは **tcp** プロファイルの設定とそれらの設定のデフォルト値を継承していますが、中には変更されている設定値もあります。**tcp-wan-optimized** プロファイルを実装することで、カスタムプロファイルを作成しなくても、いくつかの方法によりローカル TCP トラフィックのパフォーマンスを最適化することができます。

tcp-wan-optimized プロファイルをそのまま使用するか、もしくは **tcp-wan-optimized** プロファイルを親プロファイルとして別のカスタムプロファイルを作成することも可能です。

tcp-wan-optimized プロファイルのデフォルト設置値は、表 8.7 の値を除いて、**tcp** プロファイルのものと同じになっています。

設定	説明	デフォルト値
Proxy Buffer Low	受信ウィンドウが開かれたプロキシバッファレベルを指定します。	**131072**
Proxy Buffer High	受信ウィンドウが閉じられたプロキシバッファレベルを指定します。	**131072**
Send Buffer	BIG-IP にバッファサイズ（バイト単位）を送信させます。	**65535**
Receive Window	BIG-IP にウィンドウサイズ（バイト単位）を受信させます。	**65535**
Selective ACKs	この設定がチェックされている（有効になっている）場合、BIG-IP では、システムパフォーマンスを向上させるために、可能な場合は常に選択的 ACK を使用してデータを処理します。	**Enabled**（チェックあり）
Nagle’s Algorithm	チェックされている（有効になっている）場合、BIG-IP では Nagle のアルゴリズムを適用して、ネットワーク上の短いセグメントの数を減らします。デフォルトでは無効に設定されています。Telnet などの対話型プロトコルに対してこの設定を有効にした場合、高遅延ネットワークにおいて性能低下を引き起こす可能性があります。	**Enabled**（チェックあり）

表 8.7 tcp プロファイルのものと異なる tcp-wan-optimized プロファイルの設定値

UDP プロファイルの設定

UDP プロファイルは、UDP ネットワークトラフィックを管理するための設定ツールです。表 8.8 に、UDP プロファイルの設定を列挙して説明します。

設定	説明	デフォルト値
Name	この設定は、プロファイルの一意の名前を指定します。	デフォルト値なし
Parent Profile	この設定は、親プロファイルとして使用するプロファイルを指定します。新しいプロファイルは、指定された親プロファイルから非カスタム設定および値をすべて継承します。	**udp**
Idle Timeout	この設定は、接続フローが削除対象として適格になる前に、接続がアイドル状態である秒数を指定します。アイドルタイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**60**
IP ToS	この設定は、クライアントへの UDP パケット送信時に BIG-IP によって UDP パケットに割り当てられる Type of Service（ToS）レベルを指定します。	**0**
Link QoS	この設定は、クライアントへの UDP パケット送信時に BIG-IP によって UDP パケットに割り当てられる Quality of Service（QoS）レベルを指定します。	**0**
Datagram LB	この設定がチェックされている（有効になっている）場合、BIG-IP では UDP トラフィックをパケットごとにロードバランスします。	無効（チェックされていない）
Allow No Payload	この設定がチェックされている（有効になっている）場合、BIG-IP はヘッダ情報のみを含み基本データは含まないデータグラムを渡します。	無効（チェックされていない）

***表 8.8** UDP プロファイルの設定*

SCTP プロファイルの設定

BIG-IP には、Stream Control Transmission Protocol（SCTP）トラフィックの管理に使用するためのプロファイルタイプが含まれます。***Stream Control Transmission Protocol（SCTP）***とは、シグナリングデータを転送するメッセージ指向型のアプリケーション用に設計された、汎用の業界標準転送プロトコルです。SCTP では、輻輳を回避するために適切な挙動をしたり、フラッド攻撃やマスカレード攻撃に抵抗できるようになっています。

TCP と違い、SCTP では 1 つの接続内の複数のストリームをサポートすることが可能です。***TCP ストリーム***が一連のバイトを参照するのに対して、***SCTP ストリーム***は一連のメッセージを表します。

SCTP は、セッションロスの監視と 検出が必要なアプリ ケーション の転送プロト コルとしての使用が可能です。このよう なアプリ ケーション に対して、セッション障害を検出する SCTP のメカニズムは、セッションの接続性を積極的に監視します。

SCTP プロファイルの設定は、各自のニーズに合わせて調整できます。デフォルト値を持つ設定については、デフォルト設定をそのまま使用することも変更することもできます。設定の変更は、プロファイルの作成時に行うか、プロファイルの作成後にいつでも行うことができます。プロファイルの設定に関する手順については、第 5 章「*プロファイルについて*」を参照してください。

デフォルトの **sctp** プロファイルをそのまま使用することも、カスタム SCTP プロファイルを作成することもできます。表 8.9 に、SCTP プロファイルの設定を列挙して説明します。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。	デフォルト値なし
Parent Profile	親プロファイルとして使用するプロファイルを指定します。新しいプロファイルは、指定された親プロファイルから非カスタム設定および値をすべて継承します。	**tcp**
Receive Ordered	この設定が有効になっている場合、BIG-IP は上位層に順にメッセージを送信します。	**Enabled**（チェックあり）
Send Partial	この設定が有効になっている場合、BIG-IP はアプリケーションデータの一部を受け入れます。	**Enabled**（チェックあり）
TCP Shutdown	この設定が有効になっている場合、SCTP インタンスが TCP の終了をエミュレートします。上位層ユーザプロセスから SHUTDOWN メッセージを受信すると、SCTP インスタンスは SHUTDOWN チャンクを送信して、グレースフルシャットダウンを開始します。	**Enabled**（チェックあり）
Reset on Timeout	この設定が有効になっており、SCTP 接続がアイドル接続のタイムアウト値を超えた場合、接続が削除され、システムはリセットを送信します。	**Enabled**（チェックあり）
Out Streams	チャンクが要求する発信ストリームの数を指定します。	**2**
In Streams	チャンクが要求する受信ストリームの数を指定します。	**2**

***表 8.9** SCTP プロファイルの設定*

設定	説明	デフォルト値
Send Buffer	BIG-IP にバッファサイズ（バイト単位）を送信させます。	**65536**
Receive Window	送信者が通知（ACK）を受信しなくても転送可能なバイト数を指定します。	**65535**
Transmit Chunks	バッファに格納可能な転送チャンクの数を指定します。	**256**
Receive Chunks	バッファに格納可能な受信チャンクの数を指定します。	**256**
Cookie Expiration	Cookie の有効持続時間を秒単位で指定します。	**60**
Maximum Initial Retransmit Limit	接続確立を試みることができる最大回数を指定します。	**4**
Maximum Association Retransmit Limit	データ送信を試みることができる最大回数を指定します。	**8**
Proxy Buffer Low	受信ウィンドウを開くプロキシバッファレベルを指定します。	**4096**
Proxy Buffer High	受信ウィンドウを閉じるプロキシバッファレベルを指定します。	**16384**
Idle Timeout	接続が削除対象として適格になる前に、接続がアイドル状態である秒数を指定します。アイドルタイムアウト値設定の詳細については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**
Heartbeat Interval	ハートビートチャンクを送信する前の待機時間を秒単位で指定します。	**30**
IP ToS to Peer	クライアントへのパケット送信時に、BIG-IP によって SCTP パケットに割り当てられる Type of Service（ToS）レベルを指定します。	**0**
Link QoS to Peer	クライアントへのパケット送信時に、BIG-IP によって SCTP パケットに割り当てられる Quality of Service（QoS）レベルを指定します。	**0**
Secret	Cookie 認証の key-hash method authentication code（HMAC）の計算に使用する内部秘密文字列を指定します。	**default**

***表 8.9** SCTP プロファイルの設定*

9

SSL トラフィックの管理

- SSL トラフィック管理の概要
- キーおよび証明書の管理
- SSL プロファイルについて
- HTTP プロファイルの設定
- クライアントおよびサーバ認証の設定

SSL トラフィック管理の概要

BIG-IP® ローカルトラフィック管理システムには、SSL トラフィックをインテリジェントに制御するための機能がいくつか用意されています。SSL トラフィック管理機能には、次のものがあります。

- クライアントシステムと BIG-IP の間、および BIG-IP とターゲット Web サーバの間のセキュアな接続を維持するためにクライアントおよびサーバを認証する機能
- クライアントシステムおよびサーバシステムから SSL 証明書検証タスクの負荷を軽減する機能
- ヘッダ挿入や HTTP リダイレクションなどの機能を提供する iRule コマンド
- 最新の証明書失効ステータスを取得するための Online Certificate Status Protocol（OCSP）のサポート
- Light-weight Directory Access Protocol（LDAP）サーバを使用した証明書ベースのクライアント承認

SSL ネットワークトラフィックを制御できる主な方法は、クライアントまたはサーバの SSL プロファイルを設定することです。***クライアントプロファイル***は、トラフィックプロファイルの一種です。これを使用すると、BIG-IP では、完全に SSL でカプセル化されたプロトコル経由で送信される任意のクライアントリクエストを受諾および終端することができます。***サーバプロファイル***とは、BIG-IP がターゲット Web サーバへのセキュアな接続を開始できるようにするプロファイルのタイプです。

SSL トラフィックを管理するには、次の作業を完了する必要があります。

- クライアント側のセキュアな接続を終端するために、BIG-IP にキーと証明書のペアをインストールする。
- プロファイルを設定し、そのプロファイルをバーチャルサーバに関連付ける。設定ユーティリティを使用して、プロファイルを設定します。
- プロファイルをバーチャルサーバに関連付ける。

オプションとして、個々の SSL 接続を特定の方法で管理可能にする iRule を作成することもできます。詳細については、第 17 章「*iRule の記述*」を参照してください。

クライアント側およびサーバ側トラフィックの管理

BIG-IP では、クライアント側トラフィックまたはサーバ側トラフィックのどちらに対しても SSL トラフィック管理を有効にできます。

クライアント側トラフィックとは、クライアントシステムと LBIG-IP の間の接続のことです。***サーバ側トラフィック***とは、BIG-IP とターゲットサーバシステムの間の接続のことです。

- **クライアント側 SSL トラフィックの管理**
  BIG-IP によるクライアント側 SSL トラフィックの管理を有効にすると、BIG-IP では、クライアントリクエストを復号化することで着信 SSL 接続を終端します。そして、要求をクリアテキストでターゲットサーバに送信します。次に、クリアテキストの応答（Web ページなど）を取得し、要求を暗号化してから、クライアントに Web ページを戻します。SSL 接続の終端プロセス中、BIG-IP では、オプションとして、通常はターゲット Web サーバで処理される SSL 証明書検証機能のすべてを実行できます。クライアント側 SSL プロファイルの設定については、「*SSL プロファイルについて*」（9-11 ページ）を参照してください。
- **サーバ側 SSL トラフィックの管理**
  BIG-IP によるサーバ側 SSL トラフィックの管理を有効にすると、BIG-IP により、復号化された要求がターゲットサーバへの送信前に再暗号化されるので、ネットワークのセキュリティが強化されます。この再暗号化に加えて、BIG-IP では、オプションとして、クライアント証明書に対して実行できるのと同じ検証機能をサーバ証明書に対して実行できます。サーバ側 SSL プロファイルの設定については、「*SSL プロファイルについて*」（9-11 ページ）を参照してください。

SSL トラフィック制御機能の要約

BIG-IP には、SSL トラフィックの制御に関連した重要な機能がいくつか用意されています。これらのうち最も重要な機能は、SSL 認証（つまり、証明書の検証と失効）および暗号化と復号化です。これらの機能のほとんどはクライアントまたはサーバの SSL プロファイルの設定を通じて使用できますが、iRules™ などの機能を通じて提供されるものもあります。表 9.1 は、SSL トラフィック管理に関連した機能を要約します。表の次の項では、これらの機能について説明します。

機能	説明	設定ツール
証明書の検証	SSL 接続を終端および開始する際、BIG-IP では、完全な証明書の検証タスク（クライアント証明書とサーバ証明書の両方の検証を含む）を実行できます。たとえば、BIG-IP がクライアント証明書を検証する範囲を定義できます。つまり、クライアント証明書をリクエストするか、要求するか、またはすべてのクライアント証明書を無視するように BIG-IP を設定できます。さらに、証明書チェーントラバースのレベルなどの設定を指定することも可能です。	クライアントおよびサーバの SSL プロファイル iRules
証明書の失効	クライアントまたはサーバが BIG-IP に証明書を提示すると、BIG-IP は証明書の検証プロセスの一環として、証明書の失効ステータスをチェックできます。	クライアントおよびサーバの SSL プロファイル OCSP プロファイル

***表 9.1** SSL トラフィック制御に関連した BIG-IP の機能の要約*

機能	説明	設定ツール
暗号化および復号化	ターゲット Web サーバから負荷を軽減する手段の 1 つとして、BIG-IP では、着信クライアントリクエストを復号化します。追加のセキュリティオプションとして、要求がサーバに転送される前に再暗号化されるようにプロファイルを設定できます。要求および応答の暗号化と復号化は、SSL プロファイル設定の一環としてユーザが指定する特定の暗号に基づいています。	クライアントおよびサーバの SSL プロファイル
承認	BIG-IP を特定の方法で設定することで、システムリソースへのアクセスを制御できます。たとえば、クライアント証明書情報をクライアントリクエストに挿入し、その情報に基づいてアクセスを許可するための iRule を作成できます。また、LDAP データベースサーバを含む環境では、BIG-IP は、クライアント証明書を使用して LDAP サーバをクエリすることで、リソースへのアクセスを許可できます。同時 TCP 接続の数を制限することもできます。	SSL クライアント証明書 LDAP プロファイル iRules
SSL セッションパーシステンス	BIG-IP の強力な機能として、SSL セッションのパーシステンスを有効にできる機能があります。SSL 接続を終端するように BIG-IP を設定しているかどうかによって、2 つのタイプの SSL パーシステンスを利用できます。	クライアントおよびサーバの SSL プロファイル SSL パーシステンスプロファイル iRules
その他の SSL 機能	上記の機能に加えて、BIG-IP では、無効なプロトコルバージョン、SSL セッションキャッシュのサイズおよびタイムアウト値、SSL シャットダウン動作などのオプションも設定できます。	クライアントおよびサーバの SSL プロファイル

***表 9.1** SSL トラフィック制御に関連した BIG-IP の機能の要約（続き）*

証明書の検証について

証明書の検証とは、クライアントまたはサーバがピア（つまり、クライアントまたはサーバ）によって提示された証明書を信頼できるかどうかを判断するプロセスです。BIG-IP は、クライアントから要求を受信したりサーバから応答を受信すると、そのクライアントまたはサーバによって提示された証明書が信頼できるかどうかを検証しようとします。

署名付き証明書

SSL 接続を処理するクライアントおよびサーバに対する基本的なセキュリティ要件は、それぞれがピアと通信するたびに、信頼できる認証局（CA）によって署名された証明書を提示するということです。オプションとして、クライアント証明書の検証タスクを処理するように BIG-IP を設定できます。

BIG-IP 上で証明書の検証を設定する場合、SSL 要求の処理時にクライアントに提示できる証明書が BIG-IP によって保持されている必要があります。

オプションで、サーバ応答を検証するように BIG-IP を設定できます。その際、サーバによって特に BIG-IP からの証明書が要求される場合には、第2の証明書がBIG-IPによって保持されている必要があります。

クライアント証明書およびサーバ証明書を検証するように BIG-IP を設定する場合、BIG-IP による SSL トラフィック管理を可能にするには、キーおよび証明書を生成してシステムにインストールしておく必要があります。これは、設定ユーティリティを使用して行えます。新しいキーと証明書のペアを生成するには、既存のペアをインポートすることもできます。既存のキーと証明書のペアのインポートに関する詳細については、「キー、*証明書*、*およびアーカイブのインポート*」（9-10 ページ）を参照してください。

クライアント側の証明書の検証

クライアントによって HTTP リクエストが行われると、BIG-IP では、通常はターゲットサーバによって実行されるクライアント証明書の検証タスクを実行できます。

クライアント によって BIG-IP に証明書が提示されると 、BIG-IP では、***クライアントの信頼できる CA ファイル***を使用して、信頼できる認証局を判別します。BIG-IP では、クライアント 証明書を検証しようとする際、主な手段として、このファイルを使用します。クライアント 側のプロファイルを作成する際、デフォルト のクライアント の信頼できる CA ファイルが BIG-IP によって自動的に作成されます。プロファイルで指定されているデフォルト のファイル名を使用するか、別のファイル名を指定できます。詳細については、*「信頼できるクライアント CA の指定」（9-15 ページ）*を参照してください。

クライアント証明書の検証における第 2 の要素は、プロファイルで信頼される CA のリストをクライアントにアドバタイズするために BIG-IP によって使用される、***クライアント証明書 CA ファイル***です。ただし、このリストは、BIG-IP によって*実際に*信頼される CA のリストとは異なる場合があります。信頼できる CA ファイルの場合は、このファイルのデフォルトバージョンが BIG-IP によって自動的に作成されます。

また、BIG-IP によってクライアント 証明書の検証プロセスの一環としてクライアント に送信される 、クライアント チェーンファイルも存在します。デフォルト のクライアント チェーンファイルは、クライアント の信頼できる CA ファイルです。詳細については、*「信頼できるクライアント CA の指定」（9-15 ページ）*を参照してください。

サーバ側の証明書の検証

サーバ側の検証は、サーバの SSL プロファイルを有効にして、サーバ証明書の提示を [**require**] に設定している場合に行われます。

サーバによって BIG-IP に証明書が提示されると、BIG-IP では***サーバの信頼できる CA*** ファイルを使用して、信頼できる認証局を判別します。BIG-IP では、サーバ証明書を検証しようとする際、主な手段として、このファイルを使用します。サーバ側のプロファイルを設定する際、デフォルトのサーバの信頼できる CA ファイルが BIG-IP によって自動的に作成されます。プロファイルで指定されているデフォルトのファイル名を使用するか、別のファイル名を指定できます。

また、BIG-IP によってサーバ証明書の検証プロセス全体の一環としてサーバに送信される、サーバチェーンファイルも存在します。デフォルトのサーバチェーンファイルは、サーバの信頼できる CA ファイルです。

証明書の失効について

BIG-IP では、クライアントまたはサーバによって提示された証明書が失効しているかどうかを確認できます。クライアント証明書が失効している場合、BIG-IP ではそのクライアントを認証できません。

BIG-IP では、証明書の失効ステータスをチェックするために、以下に示す 2 つの業界標準の方式がサポートされています。

- **証明書失効リスト（CRL）**
  CRL は、BIG-IP に提示されている証明書が失効しているかどうかをチェックするために BIG-IP で使用できる方式です。この CRL サポートは、CRL ファイルおよび CRL パスの形式です。BIG-IP を使用すると、クライアント側のプロファイルに対して 1 つの CRL ファイルおよびパスを設定し、サーバ側のプロファイルに対して 1 つの CRL ファイルおよびパスを設定できます。CRL を使用するための設定は、SSL プロファイルを通じて行えます。詳細については、「*クライアントおよびサーバ認証の設定*」（9-25 ページ）を参照してください。
- **Online Certificate Status Protocol（OCSP）**
  CRL を使用する場合と異なり、OCSP を使用すると、証明書の失効ステータスが常に最新のものになります。OCSP の設定は、認証プロファイルを通じて行います。詳細については、第 10 章「*アプリケーショントラフィックの認証*」を参照してください。

暗号化と復号化について

BIG-IP には、通常はターゲットサーバによってクライアントリクエストの処理の一環として実行される暗号化および復号化タスクを処理できる機能もあります。BIG-IP では、クライアント側 SSL プロファイルを使用して、着信要求を復号化してから、平文でターゲットサーバに送信します。BIG-IP では、サーバ側のプロファイルを使用して、要求を再暗号化することでセキュリティレベルを追加してから、ターゲットサーバに送信します。

暗号化に関連した BIG-IP の機能には、要求を暗号化するためにクライアントによって使用される暗号を着信 HTTP リクエストに挿入できる機能があります。その後、ユーザは、その暗号仕様に基づいてトラフィックをダイレクトできます。クライアントリクエストの宛先を制御するための暗号の指定に関する詳細については、第 17 章「*iRule の記述*」を参照してください。

クライアント承認について

認証とは異なり、***承認***は、ID の信頼性ではなく、システムリソースへのアクセスの制御に関するものです。クライアントまたはサーバの信頼性が SSL プロファイルによって検証された場合、BIG-IP では、宛先コンテンツへの接続のレベルおよびアクセスのタイプを制御できるようになります。

SSL 接続に対するアクセス制御をサポートする BIG-IP の機能は、以下のとおりです。

- HTTP リクエストへのクライアント証明書フィールドの挿入
- 同時クライアント TCP 接続数の制限
- LDAP データベースサーバによるクライアント承認

システムリソースへのクライアントのアクセスを制御する最も便利な方法の 1 つは、クライアント証明書のフィールドをヘッダとしてクライアントリクエストに挿入する iRule を作成することです。たとえば、iRule は、クライアント証明書のステータスをヘッダとして要求に挿入した後、**HTTP::header** コマンドを使用して、そのステータスに基づいてターゲットサーバを選択できます。

このヘッダ挿入機能を使用したクライアントリクエストの宛先の制御に関する詳細については、第 17 章「*iRule の記述*」を参照してください。

SSL セッションパーシステンスについて

実装可能な SSL パーシステンスには、2 つのタイプがあります。1 つ目のタイプは、標準 SSL パーシステンスモードです。このモードでは、SSL 要求の復号化および SSL 応答の再暗号化を含まない SSL セッションに対してパーシステンスを有効にします。この SSL パーシステンスモードを有効にするには、SSL パーシステンスプロファイルを設定します。詳細については、第 7 章「*セッションパーシステンスの有効化*」を参照してください。

SSL パーシステンスの 2 つ目のタイプでは、要求の復号化および応答の再暗号化を含む SSL セッションに対してパーシステンスを有効にします。この場合、SSL パーシステンスの実装は、SSL セッション ID をヘッダとして HTTP リクエストに挿入することで行います。セッション ID ヘッダの挿入は、iRules を作成することで行います。iRules については、第 17 章「*iRule の記述*」を参照してください。

その他の SSL 機能について

前述の機能を使用することに加えて、無効なプロトコルバージョンの指定、SSL セッションキャッシュのサイズおよびタイムアウト値の設定、SSL シャットダウン動作の設定などの作業も実行できます。また、タイムアウト値およびデータ送信量に基づいて SSL セッションを再ネゴシエーションするように SSL プロファイルを設定することも可能です。

キーおよび証明書の管理

SSL 接続の復号化および暗号化を有効にするには、1 つ以上の SSL 証明書を BIG-IP にインストールしておく必要があります。これらの証明書の目的については、「*証明書の検証について*」（9-3 ページ）に説明があります。

証明書要求を生成して認証局に送信する作業を容易にするために、BIG-IP では、設定ユーティリティ内に一連のキー管理画面が用意されています。これらの証明書管理画面には、[Main] タブの [**Local Traffic**] セクションからアクセスできます。

キーを管理する場合、次のことが可能です。

- 既存のキーのペアおよび証明書のすべてに関する情報を表示する。
- 新しいキーのペアおよび証明書に対して要求を作成し、認証局に送信する。
- 証明書要求を更新する。
- キーおよび証明書のプロパティを表示する。
- PEM 形式のキーおよび証明書をインポートおよびエクスポートする。

既存のキーおよび証明書に関する情報の表示

既存のキーのペアおよび証明書に関する要約情報には、設定ユーティリティからアクセスできます。[**SSL Certificates**] 画面に、次の情報が表示されます。

- 証明書のステータス（有効、期限切れ間近、期限切れ）
- 要求の作成時に証明書に割り当てた一意の名前
- 証明書の発行者
- 有効期限

既存の証明書およびキーに関する情報を表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SSL Certificates**] をクリックします。
   [SSL Certificates] 画面が開き、BIG-IP にインストールされているすべての証明書がリスト表示されます。
2. [Name] カラムで、証明書名をクリックします。
   これによって証明書のプロパティが表示されます。
3. その証明書に関連付けられているキーに関する情報を表示するには、メニューバーの [**Key**] をクリックします。
   これによりキーのタイプとサイズが表示されます。

新規の証明書およびキーに対する要求の作成

設定ユーティリティを使用して、自己署名証明書（通常は内部テスト目的にのみ使用）を生成するか、または証明書とキーのペアに対する要求を作成して、認証局に送信できます。認証局に要求を送信すると、認証局によって署名付き証明書が返されます。

証明書要求を認証局に送信するには、次の 2 つの方法のいずれかを使用します。

- 設定ユーティリティの画面から新規に生成された要求のテキストをコピーし、それを認証局に送信します（カット & ペーストを使用）。
- 新規に生成された要求をファイルにダウンロードして、そのファイルを認証局に送信します。

要求を（テキストのペーストまたはファイルの添付によって）認証局に送信するには、認証局の Web サイトにアクセスします。認証局による署名を受けるために要求を送信するための設定ユーティリティの画面には、さまざまな認証局の Web サイトへのリンクが含まれています。

自己署名証明書を要求するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SSL Certificates**] をクリックします。
   [SSL Certificates] 画面が表示されます。
2. 画面右上の [**Create**] をクリックします。
3. 証明書の一意の名前を入力します。
4. [**Issuer**] 設定で [**Self**] を選択します。
5. [**Common Name**]設定およびその他の必要な設定をすべて設定します。
6. [Key Properties] セクションで、キーのサイズを選択します。
7. [**Finished**] をクリックします。

認証局から証明書を要求するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SSL Certificates**] をクリックします。
   [SSL Certificates] 画面が表示されます。
2. 画面右上の [**Create**] をクリックします。
3. 証明書の一意の名前を入力します。
4. [**Issuer**] ボックスで [**Certificate Authority**] を選択します。
5. [**Common Name**] 設定およびその他の設定を指定します。
6. [**Finished**] をクリックします。
   証明書要求が表示されます。
7. システム上のファイルに要求をダウンロードするには、次のいずれかを行います。
   - [**Request Text**] ボックスから証明書をコピーします。
   - [**Request File**] ボックスのボタンをクリックします。
8. [**Certificate Authorities**] ボックスで、認証局の名前をクリックします。
   選択した認証局の Web サイトが表示されます。

9. Web サイト上の説明に従って、コピーした要求をペーストするか、または生成された要求ファイルを添付します。
10. [**Finished**] をクリックします。

証明書の更新

証明書にはすべて、有効期限があります。証明書が期限切れになった場合、継続して使用するには、証明書を更新する必要があります。

自己署名付き証明書を更新する際、[**Lifetime**] 設定の値を変更することで、有効期限を明示的に設定することができます。この設定のデフォルト値は **365** 日です。

証明書を更新するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SSL Certificates**] をクリックします。
   既存の証明書のリストが表示されます。
2. [Name] カラムで、更新するする証明書の名前をクリックします。
   選択した証明書のプロパティが表示されます。
3. 画面の下部で [**Renew**] をクリックします。
4. 設定を変更するか、そのまま使用します。
   [**Common Name**] 設定の値は必須であることに注意してください。
5. [**Finished**] をクリックします。
6. 設定ユーティリティを閉じて、**bigpipe** シェルにアクセスします。
7. **bigpipe** シェルプロンプトで、**load** コマンドを入力します。

   *注: 証明書の更新を完了させるには、かならず **bigpipe** ユーティリティの **load** コマンドを入力する必要があります。*

証明書とキーのペアの削除

不要になった証明書とキーは削除できます。

証明書とキーのペアを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SSL certificates**] をクリックします。
これによって証明書の一覧が表示されます。
2. 削除する証明書の名前の左側にある [Select] ボックスをチェックします。
3. 画面下部の [**Delete**] をクリックします。
確認画面が表示されます。
4. [**Delete**] をクリックします。
選択した証明書が削除されます。

◆注

*SSL 証明書リスト画面で、証明書名をクリックすると、証明書のプロパティが表示されます。**[Delete]** ボタンを押せば、証明書とキーのペアの内、証明書だけを削除することができます。同様に、キーのプロパティを表示させて **[Delete]** ボタンを押せば、キーだけを削除することが可能です。証明書を完全に削除するには、上述の手順を使用してください。*

キー、証明書、およびアーカイブのインポート

キーと 証明書のペア、証明書、またはキーと 証明書のアーカイブを別のシステムから BIG-IP に転送しており 、その証明書またはアーカイブがファイルまたはベース 64エンコード テキスト 文字列の形式である場合、この証明書またはアーカイブを設定ユーティリティにインポートできます。証明書またはアーカイブを設定ユーティリティにインポート すると 、その証明書またはアーカイブを管理する作業が容易になります。インポート 対象の証明書が Privacy Enhanced Mail（PEM）形式の場合のみ、[Import SSL Certificates and Keys] 画面を使用できます。

キーのペア、証明書、またはアーカイブをインポートするには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SSL Certificates**] をクリックします。
これによって証明書の一覧が表示されます。
2. 画面の右上で [**Import**] をクリックします。
3. [**Import Type**] リストから、インポートのタイプ（[**Key**]、[**Certificate**]、または [**Archive**]）を選択します。
該当するインポートタイプの設定画面が展開されます。
4. 選択したインポートタイプに必要な設定を行います。
   - **Key** または **Certificate** を選択した場合、[**Name**] と [**Source**] を設定します。
   - **Archive**を選択した場合、[**Upload Archive File**]を設定します。

5. （キーと証明書の場合は）[**Import**] を、または（アーカイブの場合は）[**Load**] をクリックします。

アーカイブの作成

アーカイブは、1 つ以上のキーおよび証明書に対して作成できます。複数のキーおよび証明書を別のシステムにエクスポートする場合に、アーカイブを作成します。

アーカイブを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SSL Certificates**] をクリックします。
   [SSL Certificates] 画面が開きます。
2. リストの下部で [**Archive**] をクリックします。
3. アーカイブファイルの名前を入力します。
   ファイルには拡張子 **.tgz** が付きます。
4. [Key List] 領域で、次の操作を行います。
   a) [**Available Keys**] ボックスで、アーカイブするキーを選択します。
   b) [Move] ボタン（**<<**）を使用して、キー名を [**Keys to Archive**] ボックスに移動します。
5. [Certificate List] 領域で、次の操作を行います。
   a) [**Available Certificates**] ボックスで、アーカイブする証明書を選択します。
   b) [Move] ボタン（**<<**）を使用して、証明書名を [**Certificates to Archive**] ボックスに移動します。
6. [**Generate & Download Archive**] ボタンをクリックします。

SSL プロファイルについて

第 5 章「*プロファイルについて*」で説明したように、***プロファイル***とは、BIG-IP によってアプリケーション固有のネットワークトラフィックを管理する方法を決定する値を持つ設定のグループです。プロファイルで管理可能なトラフィックタイプの 1 つは、SSL トラフィックです。SSL プロファイルの最も基本的な機能は、ターゲット Web サーバから証明書の妥当性検査および検証タスクや暗号化と復号化の負荷を軽減することです。

SSL プロファイルには、次の 2 つのタイプがあります。

- **クライアントプロファイル**
  クライアントプロファイルを使用すると、BIG-IP では、クライアントシステムから BIG-IP への SSL 接続すべてに対して認証および暗号化タスクを処理できます。このタイプのプロファイルを実装

するには、デフォルトの **clientssl** プロファイルを使用するか、または **clientssl** プロファイルに基づいてカスタムプロファイルを作成します。

- **サーバプロファイル**
  サーバプロファイルを使用すると、BIG-IP では、システムからターゲットサーバに送信される SSL 接続すべてに対して暗号化タスクを処理できます。サーバ SSL プロファイルは、BIG-IP の認証が必要な場合にサーバに証明書を提示することで、クライアントとして機能するようになっています。このタイプのプロファイルを実装するには、デフォルトの **serverssl** プロファイルを使用するか、または **serverssl** プロファイルをもとにカスタムプロファイルを作成します。

◆ 重要

*SSL 処理を有効にする前に、まず信頼できる認証局からの有効な x509 証明書を生成するか、または一時的な証明書を生成して、それを BIG-IP にインストールする必要があります。証明書の詳細については、「**署名付き証明書**」(9-3 ページ）を参照してください。BIG-IP での証明書の生成およびインストール方法については、「**キーおよび証明書の管理**」(9-7 ページ）を参照してください。*

クライアントまたはサーバの SSL プロファイルを構成する設定は、3 つのカテゴリ（汎用プロパティ、設定、クライアント認証またはサーバ認証のいずれか）に分けられます。SSL プロファイルの作成時、またはプロファイルの作成後にプロファイルの設定を変更することで、これらの設定を行えます。プロファイルの設定に関する手順については、第 5 章「*プロファイルについて*」を参照してください。

◆ ヒント

*個々の SSL 接続に対して、SSL プロファイル設定の値を無効にできます。これを行うには、iRule を作成し、無効にする設定に対応する SSL iRule コマンドを含めます。詳細については、第 17 章「**iRule の記述**」と、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。*

HTTP プロファイルの設定

この項では、[Client SSL Profile] 画面または [Server SSL Profile] 画面の [Configuration] セクションに表示される設定について説明します。その他の SSL プロファイル設定については、「*クライアントおよびサーバ認証の設定*」（9-25 ページ）を参照してください。

SSL プロファイルの作成または変更手順については、第 5 章「*プロファイルについて*」を参照してください。

表 9.2 に、クライアントまたはサーバの SSL プロファイルに対して行える設定を示します。デフォルト値を持つ設定については、デフォルト設定をそのまま使用することも変更することもできます。この表の次に、これらの設定について説明します。

設定	説明	デフォルト値
Name	プロファイルのユーザ定義名を指定します。	デフォルト値なし
Parent Profile	カスタムプロファイルの派生元とするシステム提供のプロファイルを指定します。	**clientssl** または **serverssl**
Certificate	SSL 接続を終端または開始するために BIG-IP にインストールされている証明書の名前を指定します。	**default（クライアント）** **None（サーバ）**
Key	SSL 接続を終端または開始するために BIG-IP にインストールされているキーの名前を指定します。	**default（クライアント）** **None（サーバ）**
Pass Phrase	キーの暗号化に使用されるパスフレーズの名前を指定します。	デフォルト値なし
Confirm Pass Phrase	キーの暗号化に使用されるパスフレーズの名前を確認します。	デフォルト値なし
Chain	プロファイルを認証するためにクライアントで使用可能な証明書チェーンファイルを指定または作成します。詳細については、*「証明書チェーンの設定」（9-15 ページ）*を参照してください。	**None**
Trusted Certificate Authorities	BIG-IP で信頼できるクライアント CA またはサーバ CA のリストを指定することで、証明書の検証を設定します。詳細については、*「信頼できるクライアント CA の指定」（9-15 ページ）*を参照してください。	**None**
Ciphers	BIG-IP でサポートされる暗号のリストを指定します。詳細については、「*SSL 暗号の指定*」（9-16 ページ）を参照してください。	**DEFAULT**
Options	SSL 処理に関連した業界関連のさまざまな回避策のセットを有効にする [**All Bugfixes Enabled**] の値を指定します。詳細については、「*回避策の設定*」（9-17 ページ）を参照してください。	**All Bugfixes Enabled**
ModSSL Methods	ModSSL 方式のエミュレーションを有効または無効にします。この設定はデフォルトでは無効になっています。	無効（チェックされていない）
Cache Size	SSL セッションキャッシュ内のセッション数を指定します。	**20000**
Cache Timeout	SSL セッションキャッシュエントリのタイムアウト値（秒単位）を指定します。	**3600**
Alert Timeout	SSL 接続をリセットする前、接続を終端しようとする間に、BIG-IP が待機する時間を秒単位で指定します。	**60**

***表 9.2** SSL プロファイルの設定*

設定	説明	デフォルト値
Handshake Timeout	SSLハンドシェイクのタイムアウト値を秒単位で指定します。	**60**
Renegotiate Period	初期接続時間から何秒経った時点で、SSL セッションの再ネゴシエーションを行うかを指定します。	**Indefinite**
Renegotiate Size	SSL 再ネゴシエーションの指定スループットサイズ（バイト単位）を強制します。	**Indefinite**
Renegotiate Max Record Delay	SSL 再ネゴシエーションの最大レコード遅延を強制します。 **注**：[**Renegotiate Max Record Delay**] 属性は、クライアント側のプロファイルにのみ適用されます。	**10**
Unclean Shutdown	強制 SSL シャットダウンを有効または無効にするように BIG-IP を設定します。	チェックあり（有効）
Strict Resume	強制シャットダウンの後、SSL セッションの再開を有効または無効にするように BIG-IP を設定します。	チェックなし（無効）
非 SSL 接続	トラフィック管理システム内を非 SSL 接続が通過するのをクリアテキスト形式で許可または禁止します。	チェックなし（無効）

***表 9.2** SSL プロファイルの設定*

SSL プロファイルを設定する前に、変更する可能性がある特定の設定に関する説明を読んでおくと役立ちます。

プロファイル名の指定

SSL プロファイルを作成するには、プロファイルの一意の名前を指定する必要があります。[**Name**] 設定は、SSL プロファイルの作成時にアクティブに指定する必要がある唯一の設定です。その他の設定にはすべて、デフォルト値があります。

親プロファイルの選択

作成する各プロファイルは、親プロファイルから派生します。[**Parent Profile**] 設定を使用して、デフォルトの **SSL** プロファイルを親プロファイルとして設定するか、すでに作成している別の SSL プロファイルを設定することができます。

証明書の名前の指定

[**Certificate**] 設定の値は、クライアント側またはサーバ側の SSL 接続を認証するために、BIG-IP にインストールしている証明書の名前です。証明書要求を生成していない場合、または BIG-IP に証明書をインストールしていない場合には、既存の証明書の名前、**default** を指定できます。証明書の詳細については、「*証明書の検証について*」（9-3 ページ）を参照してください。

キーの名前の指定

[**Key**] 設定の値は、クライアント側またはサーバ側の SSL 接続を認証するために、BIG-IP にインストールしているキーの名前です。キー要求を生成していない場合、または BIG-IP にキーをインストールしていない場合には、既存のキーの名前、**default** を指定できます。

キーの詳細については、「*証明書の検証について*」（9-3 ページ）を参照してください。

パスフレーズの指定と確認

[**Pass Phrase**] 設定と [**Confirm Pass Phrase**] 設定で、SSL キーを暗号化するための文字列を指定することができます。この機能はオプションです。これらの設定にアクセスするには、[Configuration] リストから [**Advanced**] を選択してください。

セキュリティを高めるために、これらの設定を使用する場合は、パスフレーズ自体が自動で暗号化されるようになっています。このパスフレーズ暗号化は、BIG-IP ユーザに見えないように行われるため、有効にする必要はありません。

パスフレーズを暗号化すると、暗号化されていない状態よりも長くなります。暗号化されたパスフレーズは、**MDEyMzQ1Njc4OWFiY2RlZmdoaWprbG1ub3BxcnN0dXZ3eHl6** のようになります。

証明書チェーンの設定

クライアント検証プロセスでは、BIG-IP がクライアントを認証するだけでなく、クライアントが BIG-IP を認証する必要がある場合もあります。ただし、BIG-IP がクライアントに対して自身を認証するために使用する証明書は、そのクライアントによって信頼されていない中間 CA によって署名されていることがあります。この場合、BIG-IP は、証明書チェーンの使用を必要とすることがあります。プロファイルを使用すると、ユーザが特定の証明書チェーンファイルの名前を指定できます。ただし、チェーンファイルを構成する証明書ファイルは PEM 形式である必要があります。

［**Chain**］を設定する場合、指定可能な値は、［**None**］、［**ca-bundle**］、および［**default**］です。

信頼できるクライアント CA の指定

クライアント側の SSL 処理については、クライアントまたはサーバによって提示された証明書を検証するように SSL プロファイルを設定できます。[**Trusted Certificate Authorities**] 設定を使用して、クライアントの信頼できる CA ファイルの名前を指定できます。このファイルは、BIG-IP によって、クライアント証明書またはサーバ証明書を検証するために使用されます。信頼できる CA ファイルを設定しない場合、プロファイルではデフォルトファイルが使用されます。

ユーザが証明書の検証に対して指定する信頼できる CA ファイルには、Privacy Enhanced Mail（PEM）形式の証明書が 1 つ以上含まれます。このファイルは手動で作成されたもので、SSL プロファイルで信頼されるクライアント証明書またはサーバ証明書のリストを含んでいます。信頼できる CA ファイルを指定しない場合、または指定した信頼できる CA ファイルに BIG-IP がアクセスできない場合には、デフォルトのファイル名が使用されます。

[**Trusted Certificate Authorities**] を設定する場合、指定可能な値は、[**None**]、[**ca-bundle**]、および [**default**] です。

SSL 暗号の指定

SSL プロファイルごとに、SSL 接続で利用可能な暗号を指定できます。暗号を設定する際には、要求を送信するクライアントの暗号、または応答を送信するサーバの暗号と一致する暗号を SSL プロファイルに対して設定する必要があります。

たとえば、クライアントは、クライアント側 SSL とサーバ側 SSL の両方を使用するように設定されている SSL プロファイルに接続し、SSL 接続を正常に確立できます。クライアントが追加のデータ（HTTP リクエストなど）を送信した後、SSL プロファイルはサーバへの SSL 接続を確立しようとします。ただし、SSL プロファイルはサーバ側 SSL に対して 3DES 暗号のみを有効にするように設定されており、サーバは RC4 暗号のみを受け付けるように設定されていることがあります。この場合、共通の暗号が有効になっていないため、SSL プロファイルとサーバの間の SSL ハンドシェイクは失敗します。その結果、クライアント接続が閉じられます。クライアントがブラウザを使用している場合、ユーザは Web ページがロードに失敗したことを示すエラーメッセージを受信する可能性があります。

利用可能な SSL 暗号のリストを示す文字列を指定するか、またはデフォルトの暗号文字列、**DEFAULT** を使用することができます。クライアント SSL プロファイルについては、**DEFAULT** 暗号文字列の定義は **ALL:!SSLv2:@SPEED** です。サーバ SSL プロファイルについては、**DEFAULT** 暗号文字列の定義は **COMPAT+HW:@SPEED** です。

BIG-IP でサポートされる暗号は、以下のとおりです。

- SSLv2
  ***注**: 絶対に必要な場合を除き、SSLv2 暗号化の使用は避けることをお勧めします。*
- SSLv3
- TLSv1
- SGC/Set-up
- 標準のプロトコル拡張および暗号はすべて、RFC 2246 で規定されています。
- AES 暗号（RFC 3268 で規定）

◆注

BIG-IP では、OpenSSL バージョン 0.9.7 の暗号リスト形式をサポートしています。

◆ヒント

SSL プロファイルで暗号を指定することに加えて、HTTP リクエストヘッダに暗号仕様を挿入して、それらの暗号に基づいてトラフィックをダイレクトすることができます。詳細については、Web サイト ***http://devcentral.f5.com/Default.aspx?tabid=119*** *を参照してください。*

回避策の設定

OpenSSL では、問題回避策および SSL オプションのセットをサポートしています。これらの回避策およびオプションは、個々のクライアント側またはサーバ側 SSL プロファイルの設定として有効にできます。[**Options**] 設定のデフォルト値は [**Options List**] です。デフォルト値のままにすると、[**Don't insert empty fragments**] というオプションが 1 つだけ有効になります。

表 9.3 に、SSL プロファイルに対して設定可能な回避策およびオプションを列挙して説明します。

SSL 属性	説明
All Bugfixes Enabled	このオプションでは、この表で説明されている問題回避策のすべてを有効にします。問題のある実装との互換性が必要な場合、[**All Bugfixes Enabled**] オプションを使用して問題回避策のオプションを有効にすることは、通常は安全です。
Cipher server preference	BIG-IP が暗号を選択するとき、このオプションでは、クライアントの設定の代わりにサーバの設定を使用します。このオプションが設定されていない場合、SSL サーバは常にクライアントの設定に従います。このオプションが設定されている場合、SSLv3/TLSv1 サーバは自身の設定を使用して選択を行います。SSLv2 については、プロトコルが異なるため、サーバは自身の設定リストをクライアントに送信し、クライアントは常に暗号を選択します。
Don't insert empty fragments	このオプションでは、CBC 暗号に影響を及ぼす SSL 3.0/TLS 1.0 プロトコルの脆弱性に対する対策を無効にします。これらの暗号は、特定の問題のある SSL 実装では処理できません。このオプションには、その他の暗号を使用する接続については効果がありません。これは、**Enabled Options** リストのデフォルト値です。
Ephemeral RSA	このオプションでは、RSA 操作時に短命な（一時）RSA キーを使用します。仕様に従って、これは RSA キーを署名操作のみに使用可能な場合のみ（つまり、RSA キーの長さが制限されているエクスポート暗号の下で）行われます。このオプションを設定することで、BIG-IP では常に短命な RSA キーを使用します。このオプションによって SSL/TLS の仕様との互換性が損なわれ、クライアントとの相互運用性の問題が発生する可能性があるので、このオプションの使用はお勧めしません。EDH（短命な Diffie-Hellman）鍵交換による暗号を使用することをお勧めします。このオプションは、サーバ側 SSL では無視されます。

***表 9.3** 回避策およびその他の SSL オプション*

SSL 属性	説明
Microsoft session ID bug	このオプションでは、Microsoft® セッション ID の問題を処理します。
Netscape CA DN bug workaround	このオプションでは、不安定性に関する問題を処理します。Netscape® ブラウザ接続を受け付ける場合、クライアント証明書を要求する場合、Netscape では CA を持たない自己署名でない CA を持つ場合、ブラウザが証明書を持つ場合、システムはクラッシュまたはハングします。
Netscape challenge bug	このオプションでは、Netscape challenge の問題を処理します。
Netscape demo cipher change bug workaround	このオプションでは、SSL サーバのセッション再開動作を意図的に操作して、特定の Netscape サーバのセッション再開動作を模倣します（「**Netscape reuse cipher change bug workaround**」の説明を参照してください）。通常の使用では、このオプションはお勧めしません。また、このオプションは、サーバ側 SSL 処理では無視されます。
Netscape reuse cipher change bug workaround	このオプションでは、Netscape-Enterprise/2.01 内の問題を処理します。このオプションは、SSLv2/v3 経由で接続した後、SSLv3 経由で再接続する場合のみ表示されます。この場合は、暗号リストが変更されます。 まず、接続は RC4-MD5 暗号リストで確立されます。その後再開した場合、接続は DES-CBC3-SHA 暗号リストの使用に切り替わります。ただし、RFC 2246（セクション 7.4.1.3、cipher_suite）に従って、暗号リストは RC4-MD5 のままにする必要があります。 回避策として、DES-CBC-SHA:RC4-MD5 などの暗号リストで接続を試行します。何らかの理由で、新しい接続ではすべて RC4-MD5 暗号リストが使用されますが、再接続ではすべて DES-CBC-SHA 暗号リストが試行されます。したがって、Netscape では、再接続時に、常に暗号リスト内の最初の暗号が使用されます。
No SSLv2	SSLv2 プロトコルを使用しません。
No SSLv3	SSLv3 プロトコルを使用しません。
No session resumption on renegotiation	BIG-IP が SSL サーバとして再ネゴシエーションを実行する場合、このオプションでは常に新しいセッションを開始します（つまり、セッション再開要求は、最初のハンドシェイクでのみ受け付けられます）。このオプションは、サーバ側 SSL 処理では無視されます。
N0 TLSv1	TLSv1 プロトコルを使用しません。
Microsoft big SSLV3 buffer	このオプションでは、標準以外の SSL レコードサイズを使用する古い Microsoft® アプリケーションと通信するための回避策を有効にします。
Microsoft IE SSLV2 RSA padding	このオプションでは、標準以外の RSA キー埋め込みを使用する古い Microsoft® アプリケーションと通信するための回避策を有効にします。このオプションは、サーバ側 SSL では無視されます。
Passive close	**Default** に設定した場合、このオプションは、**All Bugfixes Enabled** オプションを有効にします。**None** に設定した場合、このオプションは、すべての回避策を無効にします。このオプションを **None** に設定することは、お勧めできません。
PKCS1 check 1	このデバッグオプションでは、特定の SSL サーバの脆弱性を検出するために SSL クライアントによって使用される PKCS1 埋め込みを意図的に操作します。通常の使用では、このオプションはお勧めしません。このオプションは、クライアント側 SSL 処理では無視されます。
PKCS1 check 2	このデバッグオプションでは、特定の SSL サーバの脆弱性を検出するために SSL クライアントによって使用される PKCS1 埋め込みを意図的に操作します。通常の使用では、このオプションはお勧めしません。このオプションは、クライアント側 SSL 処理では無視されます。

***表 9.3** 回避策およびその他の SSL オプション（続き）*

SSL 属性	説明
Single DH use	このオプションでは、一時または短命な DH パラメータの使用時に新しいキーを作成します。このオプションは、DH パラメータが strong prime を使用して生成されていない場合（たとえば、DSA パラメータの使用時）、小さいサブグループ攻撃を防御するために使用する必要があります。strong prime が使用されている場合、各ハンドシェイク中に新しい DH キーを生成する必要は必ずしもありませんが、これを行うことをお勧めします。一時または短命な DH パラメータが使用されているときは常に、[**Single DH use**] オプションを有効にする必要があります。
SSLEAY 080 client DH bug workaround	このオプションでは、正しくない Diffie-Hellman 公開値の長さを指定する古いSSLeayベースのアプリケーションと通信するための回避策を有効にします。このオプションは、サーバ側 SSL では無視されます。
SSL Ref2 reuse cert type bug	このオプションでは、SSL 再利用照明遺書タイプの問題を処理します。
TLS D5 bug workaround	このオプションでは、正しくない暗号化された RSA キーの長さを指定する古い TLSv1 対応アプリケーションと通信するための回避策を有効にします。このオプションは、サーバ側 SSL では無視されます。
TLS block padding bug workaround	このオプションでは、正しくないブロック埋め込みを使用する古い TLSv1対応アプリケーションと通信するための回避策を有効にします。
TLS rollback bug workaround	このオプションでは、バージョンロールバック攻撃の検出を無効にします。クライアント鍵交換中、クライアントは、最初の **hello** 中に送信するのと同じ許容可能な SSL/TLS プロトコルレベルに関する情報を送信する必要があります。クライアントによっては、サーバの回答に適応することで、このルールに違反することがあります。たとえば、クライアントは SSLv2 **hello** を送信して SSLv3.1（TLSv1）までを許容しますが、サーバは SSLv3 までしか認識しないとします。この場合、クライアントは同じ SSLv3.1（TLSv1）アナウンスを使用する必要があります。クライアントによっては、サーバの回答に応じて SSLv3 までレベルを下げ、バージョンロールバック保護に違反することがあります。このオプションは、サーバ側 SSL では無視されます。

***表 9.3** 回避策およびその他の SSL オプション（続き）*

プロトコルバージョンを設定する場合、BIG-IP のプロトコルバージョンがシステムのピアのプロトコルバージョンと一致するように設定する必要があります。つまり、クライアント側 SSL プロファイルで指定されたプロトコルバージョンはクライアントのプロトコルバージョンと一致し、サーバ側 SSL プロファイルで指定されたプロトコルバージョンはサーバのプロトコルバージョンと一致する必要があります。したがって、クライアント側とサーバ側両方の SSL 接続に対して、BIG-IP で許可しないプロトコルバージョンを指定できます。

3 つのプロトコルバージョン、**SSLv2**、**SSLv3**、および **TLSv1** のうち 2 つまでを無効として宣言できます。プロトコルバージョンを指定しない場合、BIG-IP ではすべての SSL プロトコルバージョンが許可されます。

◆注

最低でもプロトコルバージョンSSLv2を無効として指定することをお勧めします。

回避策オプションを指定するには、[**Options**] 設定で [**Options List**]（デフォルト値）が設定されていることを確認します。[**Options List**] 設定から、設定するオプションをすべて選択し、[**Enable**] ボタンをクリックします。SSL プロファイルの設定が終了したら、[**Finished**] ボタンをクリックします。

ModSSL 方式のエミュレーションの有効化

この設定では、ModSSL 方式のエミュレーションを有効または無効にします。この設定は、OpenSSL 方式が適さない場合に有効にします。

[**ModSSL Methods**] 設定を有効にしている場合、ModSSL オプションのいくつかをヘッダとして HTTP リクエストに挿入する **HTTP::header insert_modssl_fields** コマンドを使用して、iRule を作成できます。表 9.4 に、HTTP リクエストに挿入できるオプションを列挙します。

ヘッダタイプ	ヘッダの名前および形式	説明
証明書ステータス	SSLClientCertStatus: [status]	クライアント証明書のステータス。**[status]** の値は、**[NoClientCert]**、**[OK]**、または **[Error]** です。ステータスが **[NoClientCert]** の場合、このヘッダのみが要求に挿入されます。ステータスが **[Error]** の場合、エラーの後に数字のエラーコードが表示されます。
証明書バージョン	SSLClientCertVersion: [version]	証明書のバージョン。
証明書シリアル番号	SSLClientCertSerialNumber: [serial]	証明書のシリアル番号。
証明書の署名アルゴリズム	SSLClientCertSignatureAlgorithm: [alg]	証明書の署名アルゴリズム。
証明書の発行者	SSLClientCertIssuer: [issuer]	証明書の発行者。
証明書の有効期限	SSLClientCertNotValidBefore: [before] SSLClientCertNotValidAfter: [after]	証明書の有効期限。**[before]** の日付より前、**[after]** の日付より後はそれぞれ、証明書は有効ではありません。
証明書の所有者	SSLClientCertSubject: [subject]	証明書の所有者。
証明書の所有者の公開鍵	SSLClientCertSubjectPublicKey: [key]	公開鍵のタイプ。指定可能なタイプは、**[RSA ([size] bit)]**、**[DSA]**、または **[Unknown public key]** です。
証明書自体	SSLClientCert: [cert]	実際のクライアント証明書。
証明書の MD5 ハッシュ	SSLClientCertHash: [hash]	クライアント証明書の MD5 ハッシュ。

***表 9.4** iRules で使用可能な ModSSL オプション*

SSL セッションキャッシュの設定

SSL セッションキャッシュのタイムアウトとサイズの値を設定できます。各プロファイルで別個の SSL セッションキャッシュが保持されるので、プロファイルごとに値を設定できます。

SSL セッションキャッシュのサイズの設定

設定ユーティリティを使用して、SSL セッションキャッシュの最大サイズを指定できます。SSL セッションキャッシュのサイズのデフォルト値は、20,000 エントリです。値 **0** を指定すると、セッションキャッシュが許可されません。

プロファイルごとに、セッションキャッシュの最大サイズの値を設定します。SSL セッションキャッシュのサイズを指定するには、[**Cache Size**] 設定で、デフォルトのキャッシュサイズ値をそのまま使用するか、新しい値を入力します。

SSL セッションキャッシュのタイムアウトの設定

設定ユーティリティを使用して、ネゴシエーションされた SSL セッション ID の使用可能なライフタイム秒数を指定できます。SSL セッションキャッシュのデフォルトのタイムアウト値は、**300** 秒です。指定可能な値は、5 以上の整数です。

期限切れのセッション ID を持つ SSL セッションをクライアントが再開しようとすると、新しいセッションをネゴシエーションするように強制されます。

◆警告

クライアント側の SSL セッションキャッシュのタイムアウト値がゼロに設定されている場合、そのプロファイルのクライアントとネゴシエーションされた SSL セッション ID は、キャッシュがいっぱいになってエントリのパージが開始するまで、セッションキャッシュ内に残ります。これらのセッション ID を再利用するクライアントに対して重要なリソースが利用可能になっている場合、ゼロの値を設定すると、セキュリティ上のリスクが著しく増大する可能性があります。したがって、一般には、SSL セッションキャッシュのタイムアウトを、大幅に短い期間の 24 時間以下に設定するようにします。

SSL セッションキャッシュのタイムアウトを指定するには、[**Cache Timeout**] 設定で、デフォルト値をそのまま使用するか、リストから [**Specify**]、[**Immediate**]、または [**Indefinite**] を選択します。[**Specify**] を選択した場合、値を入力します。

アラートタイムアウト値の指定

[**Alert Timeout**] 設定は、SL 接続をリセットする前、接続を終端しようとする間に、BIG-IP が待機する時間を秒単位で指定します。この設定のデフォルトタイムアウト値は 60 秒です。アラートタイムアウトを指定するには、[**Alert Timeout**] 設定で、デフォルト値をそのまま使用するか、リストから [**Specify**] または [**Indefinite**] を選択します。[**Specify**] を選択した場合、値を入力します。

ハンドシェイクタイムアウト値の指定

[**Handshake Timeout**] 設定は、SSL ハンドシェイクを実行しようとするときに、BIG-IP が使用する時間を秒単位で指定します。この設定のデフォルトタイムアウト値は 60 秒です。アラートタイムアウトを指定するには、[**Handshake Timeout**] 設定で、デフォルト値をそのまま使用するか、リストから [**Specify**] または [**Indefinite**] を選択します。[**Specify**] を選択した場合、値を入力します。

SSL セッションの再ネゴシエーションの強制

存続時間の長い接続は、中間者攻撃を受けやすくなっています。このような攻撃を防御するために、時間またはアプリケーションサイズのいずれかに基づいて、BIG-IP で強制的に SSL セッションを再ネゴシエーションすることが可能です。また、指定した数のレコードが受信された後、BIG-IP で強制的に SSL セッションを終端させることもできます。

時間に基づいたセッションの再ネゴシエーション

[**Renegotiate Period**] 設定では、初期接続時間から何秒経った時点で SSL セッションの再ネゴシエーションを行うかを指定します。数を指定するか、もしくは **indefinite** か **default** のどちらかを選択してください。デフォルトは **indefinite** であり、これは SSL セッションの再ネゴシエーションが行われないことを意味しています。基本的には、セッションの再ネゴシエーションに成功するたびに、新しい接続が開始されるようになっています。したがって、セッションの再ネゴシエーションに成功すると、システムは一定時間内に再ネゴシエーションを再度試みます。たとえば、[**renegotiate period**] を **3600** 秒に設定すると、少なくとも1時間に1回の割合でセッションの再ネゴシエーションがトリガされます。

再ネゴシエーション期間を指定するには、[**Renegotiate Period**] 設定で、デフォルト値をそのまま使用するか、[**Specify**] を選択します。[**Specify**] を選択した場合、値を入力します。

アプリケーションデータのサイズに基づいたセッションの再ネゴシエーション

[**Renegotiate Size**] 設定を使用すると、指定したメガバイト数のアプリケーションデータがセキュアなチャンネルを介して送信された後、BIG-IP によって強制的に SSL セッションが再ネゴシエーションされます。この設定のデフォルト値は **Indefinite** です。

再ネゴシエーションサイズを指定するには、[**Renegotiate Size**] 設定で、デフォルト値をそのまま使用するか、新しい値を入力します。

最大レコード遅延の指定

クライアントによる再ネゴシエーションの開始を BIG-IP が待機している間、[**Renegotiate Max Record Delay**] 設定を使用すると、指定した最大数の SSL レコードが受信された後、BIG-IP によって強制的に

SSL セッションが終了されます。BIG-IP で最大数を超える SSL レコードが受信された場合、接続は閉じられます。秒単位によるこの設定のデフォルト値は **10** です。

最大レコード遅延を指定するには、**[Renegotiate Max Record Delay]** 設定で、デフォルト値をそのまま使用するか、新しい値を入力します。

SSL シャットダウンの設定

SSL 接続のシャットダウンについては、BIG-IP 上で 2 つの設定、**[Unclean Shutdown]** および **[Strict Resume]** を指定できます。

強制 SSL シャットダウンの無効化

*強制シャットダウン*では、必要な SSL シャットダウンアラートを交換せずに、基礎となる TCP 接続が閉じられます。ただし、**[Unclean Shutdown]** 設定を使用して、強制シャットダウンを無効にし、すべてのSSL接続の正常なシャットダウンが実行されるようにSSLプロファイルを強制することが可能です。

この機能は特に、Internet Explorer ブラウザに関して使用すると便利です。ブラウザのバージョンが異なる場合、また同じバージョンのブラウザでもビルドが異なる場合には、シャットダウンアラートの処理方法も異なります。バージョンやビルドによっては、サーバからのシャットダウンアラートが必要なものと不要なものがあり、SSL プロファイルではこの要件または欠落を検出できない場合があります。シャットダウンアラートがブラウザで期待されており、SSL プロファイルで交換されていない場合（デフォルト設定）、ブラウザにエラーメッセージが表示されます。

デフォルトでは、この設定は有効になっています。すなわち、BIG-IP によってすべての SSL 接続の強制シャットダウンが実行されます。強制シャットダウンを無効にするには、**[Unclean Shutdown]** 設定でチェックボックスをクリアします。

SSL セッションの中止

[Strict Resume] 設定を使用して、強制シャットダウンの後、SSL セッションを中止するように BIG-IP を設定できます。デフォルトでは、この設定は無効になっており、強制シャットダウンの後、BIG-IP によって SSL セッションが再開されます。この設定を有効にしている場合、強制シャットダウンの後、BIG-IP によって SSL セッションは再開されません。

強制シャットダウンの後に SSL セッションを中止するには、**[Strict Resume]** 設定でボックスをチェックします。

非 SSL 接続の受け入れ

[**Non-SSL Connections**] 設定を使用して、SSL 接続ではない接続を受け入れるように BIG-IP を設定できます。この場合、接続は、クリアテキスト形式で BIG-IP を通過します。デフォルトでは、この設定は無効になっています。

クライアントおよびサーバ認証の設定

この項では、[Client SSL Profile] 画面の [Client Authentication] セクション、または [Server SSL Profile] 画面の [Server Authentication] セクションに表示される設定について説明します。その他の SSL プロファイル設定については、「*HTTP プロファイルの設定*」(9-13 ページ）を参照してください。

SSL プロファイルの作成または変更手順については、第 5 章「*プロファイルについて*」を参照してください。

表 9.5 に、クライアントまたはサーバの SSL プロファイルの認証設定を列挙して説明します。デフォルト値を持つ設定については、デフォルト設定をそのまま使用することも変更することもできます。この表の次に、設定について説明します。

設定	説明	デフォルト値
Client or Server Certificate	クライアントまたはサーバによって提示される証明書をリクエスト、要求、または無視するように SSL プロファイルを設定します。	**Ignore**
Frequency	プロファイルによるクライアントの認証を、セッションごとに 1 回行うか、セッションごとに 1 回と以降に SSL セッションが再利用されるたびに 1 回行うかを指定します。詳細については、「*セッションごとの認証の設定*」(9-27 ページ）を参照してください。 ***注**: この設定は、**[Client Certificate]** を **[Request]**、**[Require]**、または **[Auto]** に設定した場合、または **[Server Certificate]** を **[Require]** に設定した場合にのみ表示されます。*	**Once**
Certificate Chain Traversal Depth	クライアント証明書チェーンでトラバースできる証明書の最大数を指定します。詳細については、「*認証レベルの設定*」(9-29 ページ）を参照してください。 ***注**: この設定は、**[Client Certificate]** を **[Request]**、**[Require]**、または **[Auto]** に設定した場合、または **[Server Certificate]** を **[Require]** に設定した場合にのみ表示されます。*	**9**
Advertised Certificate Authorities	プロファイルで信頼されている CA としてクライアントにアドバタイズする CA を指定します。詳細については、「*信頼できるクライアント CA のリストのアドバタイズ*」(9-28 ページ）を参照してください。この属性は、クライアント側のプロファイルにのみ適用されます。 ***注**: この設定は、**[Client Certificate]** を **[Request]**、**[Require]**、または **[Auto]** に設定した場合にのみ表示されます。*	**None**
Authenticate Name	サーバ証明書に埋め込まれている Common Name（CN）に基づいてターゲットサーバを認証します。詳細については、「*名前ベースの認証の設定*」(9-29 ページ）を参照してください。この属性は、サーバ側のプロファイルにのみ適用されます。 ***注**: この設定は、**[Server Certificate]** を **[Require]** に設定している場合にのみ表示されます。*	デフォルト値なし

***表 9.5** SSL プロファイル設定の認証*

設定	説明	デフォルト値
Certificate Revocation List (CRL)	失効したクライアント証明書のリストを保持することで、証明書の失効を設定します。詳細については、「*証明書の失効*」（9-29 ページ）を参照してください。 ***注**: この設定は、**[Client Certificate]** を **[Request]**、**[Require]**、または **[Auto]** に設定した場合、または **[Server Certificate]** を **[Require]** に設定した場合にのみ表示されます。*	デフォルト値なし

***表 9.5** SSL プロファイル設定の認証*

Configuring certificate presentation

[Client Certificate] または **[Server Certificate]** を設定することで、BIG-IP によってクライアントまたはサーバの認証が特定の方法で処理されるようにできます。

クライアント側の処理については、**[Client Certificate]** に指定可能な値は次のとおりです。

- **Request**
  クライアント証明書をリクエストおよび検証します。この場合、証明書のステータスまたは欠落にかかわらず、SSL プロファイルは常にアクセスを許可します。
- **Require**
  アクセスを許可する前に、有効かつ信頼できる証明書の提示をクライアントに要求します。
- **Ignore**
  証明書（または証明書の欠落）を無視するため、クライアントの認証は行いません。**[Ignore]** 設定はデフォルト設定であり、これを使用すると、セッションごとの認証設定はすべて無視されます。セッションごとの認証の設定については、「*セッションごとの認証の設定*」（9-27 ページ）を参照してください。
- **Auto**
  認証モジュールによってリクエストされるまで、クライアント証明書を無視します。この場合、オプションが **[Request]** に設定されている場合と同様に、BIG-IP によってセッション中の SSL ハンドシェイクが開始されます。この設定は、クライアント証明書の提示が必要ではない接続に対してのみ、お勧めします。

◆警告

*LDAP ベースのクライアント承認機能を使用している場合、**[Request]** または **[Ignore]** オプションを使用すると、接続が終端することがあります。LDAP ベースのクライアント承認の詳細については、第 10 章「**アプリケーショントラフィックの認証**」を参照してください。*

◆ヒント

*[**Request**] 値は、ヘッダ挿入機能と併用すると便利です。クライアント証明書情報を HTTP クライアントリクエストに挿入し、[**Request**] オプションに基づいてクライアントを認証するようにSSLプロファイルを設定している場合、BIG-IP またはサーバでは、要求を別のサーバにリダイレクトする、別のコンテンツをクライアントに返信する、クライアント証明書またはセッション ID パーシステンスを実行するなどの処理を行えるようになります。*

サーバ側の処理については、[**Server Certificate**] に指定可能な値は次のとおりです。

- **Require**
  アクセスを許可する前に、有効かつ信頼できる証明書の提示をサーバに要求します。
- **Ignore**
  証明書（または証明書の欠落）を無視するため、サーバの認証は行いません。[**Ignore**] 値はデフォルト設定であり、これを使用すると、セッションごとの認証設定はすべて無視されます。セッションごとの認証の設定については、次の「*セッションごとの認証の設定*」を参照してください。

セッションごとの認証の設定

[**Frequency**] 設定では、認証を SSL セッションごとに 1 回（**once**）、または以降に SSL セッションが再利用されるたびに 1 回（**always**）必要とするように SSL プロファイルを設定できます。このオプションのデフォルト設定は、[**once**] です。

この値を [**once**] または [**always**] のどちらに設定するかは、アプリケーションによって異なります。適切に設計されたWebアプリケーションでは、証明書の検証をセッションごとに 1 回だけ行う必要があります。パフォーマンス上の理由から、可能な場合は常にデフォルト設定（**once**）を使用することをお勧めします。

SSL プロファイルを変更することで、セッションごとに 1 回だけではなく、以降に SSL セッションが再利用されるたびに認証が必要となるように設定できます。

信頼できるクライアント CA のリストのアドバタイズ

クライアント側のプロファイルに対してのみ、認証のためにクライアント証明書を要求またはリクエストするように SSL プロファイルを設定する場合、サーバが信頼する可能性のある CA のリストをクライアントに送信するようにプロファイルを設定できます。これは、**[Advertised Certificate Authorities]** を設定することで実行できます。

このリストは、クライアント証明書 CA ファイルとも呼ばれており、クライアントの信頼できる CA ファイルとは異なります。これは、ときには、有効なクライアント証明書を所持しないクライアントがあり、その場合にプロファイルで信頼される実際の CA リストの開示を希望しないことがあるためです。クライアント証明書 CA ファイルを使用すると、証明書の検証の一環として設定された実際のクライアントの信頼できる CA ファイルとは異なる CA のリストをプロファイルがアドバタイズできるようにすることで、この問題が解決されます。

◆ ヒント

クライアント証明書 CA ファイルの内容は、クライアントの信頼できる CA ファイルの内容とは異なる場合があります。ただし、互換性の理由から、クライアント証明書 CA オプションを、実際のクライアントの信頼できる CA ファイルと一致するように設定することが最善の方法です。これは、クライアント証明書を要求するピアによって信頼できる CA のリストが提供されない場合、最近のブラウザでは SSL セッションのネゴシエーションの続行が許可されないことがあるためです。

このリストが送信されるようにプロファイルを設定するには、クライアント認証に対してサーバが信頼する 1 つ以上の CA を含む PEM 形式の証明書ファイルを指定します。クライアント証明書 CA ファイルを指定しない場合、信頼できる CA のリストはクライアントに送信されません。

◆ 注

*BIG-IP で許可されるネイティブ SSL ハンドシェイクメッセージの最大サイズは、14304 バイトです。このため、SSL ハンドシェイクがネイティブ暗号をネゴシエートし、ハンドシェイク中の全メッセージの合計長がこのバイトしきい値を超える場合、ハンドシェイクは失敗します。通常の使用では、メッセージ長がこのしきい値を超えることはありませんが、クライアント証明書を要求または要請するためにクライアント SSL プロファイルを設定する場合、**[Advertised Certificate Authorities]** 設定に多数の証明書を指定しないことをお勧めします。これにより、クライアント SSL ハンドシェイク中に交換する証明書の数を最小限に抑えることができます。*

認証レベルの設定

[**Certificate Chain Traversal Depth**] 設定を使用すると、証明書チェーン内でトラバースできる証明書の最大数を設定できます。デフォルト値は **9** です。より長いチェーンが提供されており、クライアントがこの数のトラバース内で認証されていない場合、クライアント証明書またはサーバ証明書の検証は失敗します。認証レベルの値がゼロに設定されている場合、クライアント証明書またはサーバ証明書、およびチェーンファイルの 1 つのみが検査されます。

名前ベースの認証の設定

BIG-IP では、サーバ側のプロファイルに対してのみ、中間者攻撃に対して防御する名前ベースの認証をサポートしています。サーバ側のプロファイルに [**Authenticate Name**] を設定している場合、ターゲットサーバが BIG-IP に提示する証明書にリストされている Common Name（CN）に対して名前が照合されます。指定した名前の属性がサーバ証明書の CN と一致しない場合、BIG-IP によって接続が閉じられます。CN の例としては、**www.f5.com** があります。

証明書の失効

[**Certificate Revocation List (CRL)**] 設定を使用すると、BIG-IP では、CRL を使用して、クライアントまたはサーバの認証を行う前に、証明書の失効ステータスをチェックできます。

SSL プロファイルに対して CRL を設定するには、失効したクライアント証明書またはサーバ証明書のリストを含む CRL ファイルを設定する必要があります。失効した証明書のリストを指定する場合は、PEM 形式のファイルを指定する必要があります。

◆ 重要

*CRL ファイルは古くなることがあり、1 ～ 30 日に 1 回は更新する必要があります。CRL ファイルが古い場合、BIG-IP によってすべての証明書（有効な証明書と無効な証明書の両方）が拒否されます。このため、CRL ファイルを常に最新のものにしておくことが重要です。これを行うには、**/config/ssl/ssl.crl** ディレクトリの CRL にアクセスして、**openssl crl** コマンドを使用します。詳細については、**http://www.openssl.org/docs/** を参照してください。*

CRL を使用する代わりに、Online Certificate Status Protocol（OCSP）機能を使用すると、証明書の失効ステータスに関する最新の情報を確実に得ることができます。詳細については、第 10 章「*アプリケーショントラフィックの認証*」を参照してください。

10

アプリケーショントラフィックの認証

- リモート認証の概要
- LDAP 認証モジュールの実装
- RADIUS 認証モジュールの実装
- TACACS+ 認証モジュールの実装
- SSL クライアント証明書 LDAP 認証モジュールの実装
- SSL OCSP 認証モジュールの実装
- CRLDP 認証モジュールの実装
- Kerberos 委任認証モジュールの実装

リモート認証の概要

BIG-IP® のローカルトラフィック管理システムの重要な特徴は、Pluggable Authentication Module（PAM）テクノロジをサポートしている点です。PAM テクノロジによって、さまざまな認証と承認スキームから、ネットワークトラフィックの認証や承認に最適なものを選択できます。

PAM テクノロジの目標は、BIG-IP などのアプリケーションを、基幹認証テクノロジから切り離すことです。これによって、BIG-IP が受信するアプリケーショントラフィックを認証するときに、BIG-IP で使用する特定の認証 / 承認テクノロジを指定できます。

このため、BIG-IP には***認証モジュール***と呼ばれる複数の認証スキームが用意されています。これらの***認証モジュール***を使用すると、BIG-IP を経由するアプリケーション要求をリモートシステムを使用して認証または承認できます。

認証モジュールを実装するには、設定ユーティリティを使用してユーティリティの [Local Traffic] セクション内のプロファイル画面にアクセスします。これらの画面では、実装する認証モジュールの種別を設定できます。

◆注

*通常 BIG-IP では、管理インターフェイスではなく、Traffic Management Microkernel（TMM）スイッチインターフェイス（つまり、VLAN とセルフ IP アドレスに関連付けられたインターフェイス）を介してリモート認証トラフィックをルーティングします。したがって、何らかの理由で TMM サービスが停止した場合、サービスが復旧するまでリモート認証は利用できません。BIG-IP インターフェイス経由のトラフィックの詳細は、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

BIG-IP 認証モジュール

リモート認証のために実装できる BIG-IP モジュールは以下のとおりです。

- **LDAP（Lightweight Directory Access Protocol）**
  BIG-IP は、リモート LDAP サーバまたは Microsoft® Windows® Active Directory® サーバに格納されているデータを使用してネットワークトラフィックを認証または承認します。クライアントの資格情報はベーシック HTTP 認証（ユーザ名とパスワード）に基づきます。
- **Remote Authentication Dial-In User Service (RADIUS)**
  BIG-IP はリモート RADIUS サーバに格納されたデータを使用してネットワークトラフィックを認証します。クライアントの資格情報はベーシック HTTP 認証（ユーザ名とパスワード）に基づきます。
- **TACACS+**
  BIG-IP はリモート TACACS+ サーバに格納されたデータを使用してネットワークトラフィックを認証します。クライアントの資格情報はベーシック HTTP 認証（ユーザ名とパスワード）に基づきます。

- **SSL クライアント証明書 LDAP**
  BIG-IP はリモート LDAP サーバに格納されたデータを使用してネットワークトラフィックを承認します。クライアントの資格情報は SSL 証明書、および定義済みのユーザグループとロールに基づきます。
- **Online Certificate Status Protocol（OCSP）**
  BIG-IP はリモート OCSP サーバに格納されたデータを使用してクライアント証明書の失効を確認します。クライアントの資格情報は SSL 証明書に基づきます。

認証モジュールの実装

BIG-IP を使用すると、前述の認証モジュールをどれでも実装できます。認証モジュールの実装では、次のオブジェクトを作成または設定する必要があります。

- **RADIUS サーバまたは SSL OCSP レスポンダオブジェクト**
  これは RADIUS および SSL OCSP 認証モジュールのみに必要です。*サーバオブジェクト*と*レスポンダオブジェクト*は、リモート RADIUS サーバまたは OCSP レスポンダに関する設定で構成されています。
- **構成オブジェクト**
  *構成オブジェクト*は、認証モジュールを制御する構成可能な設定のグループです。構成オブジェクトの設定には、**Hosts**、**Service Port**、**Search Time Limit** などがあります。
- **認証プロファイル**
  *認証プロファイル*は、実装する認証モジュールの種別、親プロファイル、および構成オブジェクトを指定するオブジェクトです。BIG-IP が各種認証モジュールに提供するデフォルトのプロファイルを使用することも、カスタムプロファイルを作成することもできます。プロファイルの詳細については、第 5 章「*プロファイルについて*」を参照してください。
- **認証 iRule**
  BIG-IP は、実装されるどんな種類の PAM モジュールについても、対応したデフォルトの認証 iRule を提供します。この iRule はプロファイルとバーチャルサーバに関連付ける必要があります。***認証 iRule*** は、TCL ベースのスクリプトで、対応するプロファイルと構成オブジェクトの設定に基づいて認証 / 承認アクションを実行します。

  ほとんどの場合は、カスタム iRule を作成しなくてもデフォルトの iRule を利用できます。複数の LDAP 認証モジュールなど、構成オブジェクトとプロファイル設定が異なる同じ種類の認証モジュールを複数実装する場合、カスタム iRules™ を作成する必要があります。カスタム iRule を作成する場合は、第 17 章「*iRule の記述*」を参照してください。

認証モジュールの実装に使用するプロセスは単純なものです。たとえば、ユーザ名とパスワードの資格情報を使用した認証クライアントトラフィックへのリモート LDAP サーバを使用する LDAP 認証モジュールを実装するには、次のタスクを行ってください。この例ではカスタムのプロファイルを作成し、BIG-IP が提供するデフォルトの iRule を使用しています。

認証モジュールを実装するには

1. 認証構成オブジェクトを作成し、**my_ldap_config** という名前を付けます。LDAP 認証構成オブジェクトの作成手順については、「*LDAP 構成オブジェクトの作成*」（10-5 ページ）を参照してください。
2. カスタム認証プロファイルを作成して **my_ldap_profile** という名前を付けます。そのファイル内で認証モジュールの種別を **LDAP** と指定し、親プロファイル（デフォルトの **ldap** プロファイルまたは作成した別のカスタムプロファイル）を定義します。次に構成オブジェクト **my_ldap_config** を参照します。LDAP プロファイルの作成手順については、「*LDAP プロファイルの作成*」（10-7 ページ）を参照してください。
3. カスタムプロファイルである **my_ldap_profile** とデフォルトの認証 iRule である **auth_ldap** の両方を参照するバーチャルサーバを設定します。バーチャルサーバ設定を設定する方法については、第 2 章「*バーチャルサーバの設定*」を参照してください。

LDAP 認証モジュールの実装

LDAP 認証モジュールとは、BIG-IP 経由のクライアント接続を認証または承認するための機構です。このモジュールは、認証または承認するデータがリモート LDAP サーバや Microsoft® Windows Active Directoryサーバに格納されている場合で、ベーシックHTTP認証（ユーザ名とパスワード）に基づいた資格情報が必要な場合に有効です。

LDAP 認証モジュールを使用すると、BIG-IP は、認証が成功または失敗したことや、LDAP サーバで何らかの資格情報が必要であることを表示できます。

また、システムは、サーバが LDAP クエリ応答で返した情報に基づいて、何らかの措置を取ることができます。たとえば、LDAP 応答情報では、ユーザのグループメンバシップや、ユーザのパスワードの期限切れなどを示すことができます。BIG-IP が LDAP 応答で特定のデータを返すように設定するには、コマンド **AUTH::subscribe**、**AUTH::unsubscribe**、および **AUTH::result-data** を使用して、iRule を作成します。詳細については、第 17 章「*iRule の記述*」の章と、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。

LDAP 認証モジュールを実装するには、リモート LDAP サーバ上のデータにアクセスするように BIG-IP を設定する必要があります。この作業は、次の手順で進めます。

- LDAP 構成オブジェクト
- LDAP プロファイル

◆重要

*LDAP オブジェクトとプロファイルが作成されると、BIG-IP はそれらを現在の管理パーティションに配置します。デフォルトプロファイルは常に **Common** パーティションに置かれます。パーティションの詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

これらのオブジェクトを作成した後、LDAP プロファイルをバーチャルサーバに割り当てる必要があります。

◆注

*LDAP 認証オブジェクトの作成時に、設定ユーティリティの **[Create]** ボタンをクリックできない場合は、LDAP 認証オブジェクトを作成する権限が現在のユーザロールに付与されていないことを示します。*

LDAP 構成オブジェクトの作成

LDAP 構成オブジェクトを作成する場合、さまざまな設定を行う必要があります。表 10.1 は、LDAP 構成オブジェクトの設定項目と値を示します。この表では、[New Authentication Configuration] 画面と同じ分類で設定項目を分けています。このオブジェクトを作成する詳細な手順については、「*LDAP 構成オブジェクトを作成するには*」（10-6 ページ）を参照してください。

設定	説明	デフォルト値
汎用プロパティ		
Name	構成オブジェクトの名前を一意で指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **LDAP** と指定してください。	デフォルト値なし
設定		
Remote LDAP Tree	検索ベースの識別名を指定します。	デフォルト値なし
Hosts	BIG-IP が認証データの取得に使用する LDAP サーバのアドレスを一覧します。	デフォルト値なし
Service Port	LDAP サービスのポート番号を指定します。	**389**（SSL 非対応） **636**（SSL 対応）
LDAP Version	LDAP アプリケーションのバージョン番号を指定します。	**3**
Bind DN	検索を実行するためにバインドするアカウントの識別名を指定します。この*検索*アカウントは、検索を実行するために使用される読み取り専用アカウントです。**管理者**アカウントも*検索*アカウントとして利用できます。管理者の識別名が指定されていない場合、バインドは試行されません。サイトで匿名での検索が許可されていない場合のみ、この設定を行ってください。 リモート サーバが Microsoft® Windows Active Directory サーバである場合、識別名は電子メールアドレスの形式にする必要があります。	デフォルト値なし
Bind Password	LDAP サーバ上に作成される*検索*アカウントのパスワードを指定します。この設定は **bind** DN を使用している場合に必要です。	デフォルト値なし
Confirm Bind Password	**バインド**識別名のパスワードを確認します。この設定はオプションです。	デフォルト値なし
Search Time Limit	検索タイムアウトまでの時間を指定します。	**30**
Bind Time Limit	バインドタイムアウトまでの時間を指定します。	**30**
Filter	フィルタを指定します。この設定はクライアントトラフィックの承認に使用されます。	デフォルト値なし
Login Attribute	ログイン属性を指定します。通常は **uid** を指定しますが、サーバが Microsoft Windows Active Directory サーバの場合は、アカウント名 **sAMAccountName**（大文字小文字の区別なし）を指定します。	デフォルト値なし

***表 10.1** LDAP 構成オブジェクトの設定項目*

設定	説明	デフォルト値
Group DN	グループの識別名を指定します。この設定はクライアントトラフィックの承認に使用されます。	デフォルト値なし
Group Member Attribute	グループメンバの属性を指定します。この設定はクライアントトラフィックの承認に使用されます。	デフォルト値なし
SSL	指定できる値は、**[Enabled]**、**[Disabled]**、および **[Start TLS]** です。**[Enabled]** を指定した場合、BIG-IP はサービスポート番号を **389** から **636** に変更します。	**Disabled**
Check SSL Peer	[Enabled] を指定した場合、SSL ペアがチェックされます。	**Disabled**
SSL CA Certificate	SSL CA 証明書の名前を指定します。指定できる値は、**[None]**、**[Default]**、および **[Specify Full Path]** です。	**None**
SSL Client Key	SSL クライアントキーの名前を指定します。指定できる値は、**[None]**、**[Default]**、および **[Specify Full Path]** です。	**None**
SSL Client Certificate	SSL クライアント証明書の名前を指定します。指定できる値は、**[None]**、**[Default]**、および **[Specify Full Path]** です。	**None**
SSL Ciphers	SSL 暗号を指定します。	デフォルト値なし
Ignore Unknown User	有効にすると、BIG-IP は不明なユーザを無視します。	**Disabled**
Warning Logging	警告メッセージを有効または無効にします。	**Enabled**
Debug Logging	SYSLOG デバッグ情報を LOG_DEBUG レベルで有効または無効にします。通常の使用ではお勧めしません。	**Disabled**

***表 10.1** LDAP 構成オブジェクトの設定項目（続き）*

LDAP 構成オブジェクトを作成するには

1. ナビゲーションペインの [Main] タブで、**[Local Traffic]** を展開して **[Profiles]** をクリックします。
   [Profiles] 画面が開きます。
2. **[Authentication]** メニューの **[Configurations]** を選択します。
3. 画面の右上で **[Create]** をクリックします。
   [New Configuration] 画面が表示されます。
4. **my_ldap_config** など、構成オブジェクトの一意の名前を **[Name]** 設定に入力します。

   ***注**: 名前には小文字のみを使うようにしてください。*
5. **[Type]** 設定から **[LDAP]** を選択します。
   画面が拡張され、いくつかの設定が表示されます。
6. ここで表示されるすべての設定の値を必要に応じて変更できます。
   （高度な設定を行うには、**[Configuration]** の **[Advanced]** を選択します）
7. **[Finished]** をクリックします。

LDAP プロファイルの作成

LDAP 構成オブジェクトを作成した後、LDAP プロファイルを作成または設定する必要があります。これはデフォルトの **ldap** プロファイルを変更するか、デフォルトのプロファイル設定を継承したカスタムプロファイルを作成して行います。認証プロファイルの重要な機能は、ユーザが作成した既存の構成オブジェクトを参照するということです。

ほとんどの場合、デフォルトのプロファイルをそのまま利用できます。ただし、デフォルトのプロファイルを使用した場合でも、作成した対応する構成オブジェクトを指定するようにデフォルトのプロファイルを変更する必要はあります。

カスタムプロファイルを作成する場合は、新しいプロファイルで継承する値を含んだ親プロファイル（カスタムプロファイルまたはデフォルトプロファイルのいずれか）を指定する必要があります。

LDAP プロファイルを作成する場合、いくつかの設定を行う必要があります。表 10.2 は、LDAP プロファイルを構成する設定項目と値を示します。LDAP プロファイルの作成手順については、次の「*デフォルトの LDAP プロファイルを変更するには*」、または「*カスタム LDAP プロファイルを作成するには*」（10-8 ページ）を参照してください。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **LDAP** と指定してください。	デフォルト値なし
Parent Profile	値を継承するプロファイルを指定します。	**ldap**
Mode	プロファイルが有効か無効かを指定します。指定できる値は、[**Auto**]、[**Enabled**]、および [**Disabled**] です。	**Enabled**
Configuration	既存の LDAP 構成オブジェクトを指定します。この設定は必須です。	デフォルト値なし
Rule	既存の認証 iRule の名前を指定します。iRule を指定しない場合、BIG-IP は対応するデフォルト iRule を使用します。	**auth_ldap**
Idle Timeout	認証または承認要求がタイムアウトになるまでの待ち時間を秒数で指定します。デフォルトの値を使用するか、別の値を指定できます。または **Indefinite** を選択することもできます。タイムアウト値の概要については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**

***表 10.2** LDAP プロファイルの設定*

デフォルトの LDAP プロファイルを変更するには

1. ナビゲーションペインの [Main] タブで [**Local Traffic**] を展開し、[**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Profiles**] を選択します。
   デフォルトの認証プロファイルが一覧表示されます。
3. [Name] カラムの [**ldap**] をクリックします。

4. [**Mode**] 設定から [**Enabled**] または [**Auto**] を選択します。
5. [**Configuration**] 設定のリストから構成オブジェクトを選択します。[**None**] は許可されていない値です。
6. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_ldap** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_ldap** を使用しない場合は、作成した既存の iRule の名前を選択してください。
7. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
8. [**Finished**] をクリックします。

カスタム LDAP プロファイルを作成するには

1. ナビゲーションペインの [Main] タブで [**Local Traffic**] を展開し、[**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Profiles**] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [New Profile] 画面が表示されます。
4. **my_ldap_profile** など、プロファイルの一意の名前を [**Name**] 設定に入力します。
   注: 名前には小文字のみを使うようにしてください。
5. [**Type**] 設定から [**LDAP**] を選択します。
   画面が拡張され、追加の設定が表示されます。
6. [**Parent Profile**] 設定では、プロファイルのタイプを指定します。
   - 親プロファイルにデフォルトのプロファイル **ldap** を使用する場合は、設定はそのままにします。
   - 親プロファイルにカスタムプロファイルを使用する場合は、リストからカスタムプロファイル名を選択します。
7. [**Mode**] 設定から [**Enabled**] または [**Auto**] を選択します。
8. [**Configuration**] 設定のリストから構成オブジェクトを選択します。
9. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_ldap** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_ldap** を使用しない場合は、作成した既存の iRule の名前を選択してください。
10. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
11. [**Finished**] をクリックします。

LDAP 構成オブジェクトと LDAP プロファイルを作成した後、バーチャルサーバの [**Authentication Profile**] 設定を指定して、このプロファイルをバーチャルサーバに割り当てる必要があります。バーチャルサーバ設定を設定する方法については、第 2 章「*バーチャルサーバの設定*」を参照してください。

RADIUS 認証モジュールの実装

RADIUS 認証モジュールとは、BIG-IP 経由のクライアント接続を認証するための機構です。このモジュールは、リモート RADIUS サーバに認証データが格納されている場合に使用します。この場合、クライアントの資格情報はベーシック HTTP 認証（つまりユーザ名とパスワード）に基づきます。

RADIUS 認証モジュールを実装するには、リモート RADIUS サーバ上のデータにアクセスするようにBIG-IPを設定する必要があります。この作業は、次の手順で進めます。

- 1 つ以上の上位 RADIUS サーバオブジェクトを作成する
- RADIUS 構成オブジェクトを作成する
- RADIUS プロファイルを作成する

◆重要

RADIUS オブジェクトとプロファイルが作成されると、BIG-IP はそれらを現在の管理パーティションに配置します。デフォルトプロファイルは常に ***Common*** *パーティションに置かれます。パーティションの詳細については、『****TMOS™ Management Guide for BIG-IP® Systems****』を参照してください。*

これらのオブジェクトを作成した後、RADIUS プロファイルをバーチャルサーバに割り当てる必要があります。

◆注

*RADIUS認証オブジェクトの作成時に、設定ユーティリティの****[Create]*** *ボタンをクリックできない場合は、RADIUS 認証オブジェクトを作成する権限が現在のユーザロールに付与されていないことを示します。*

RADIUS サーバオブジェクトの作成

RADIUS サーバオブジェクトを作成する場合、いくつかの設定を行う必要があります。表 10.3 は、RADIUS サーバオブジェクトを構成する設定項目と値を示します。サーバオブジェクトを作成する詳細な手順については、「*RADIUS サーバオブジェクトを作成するには*」（10-11 ページ）を参照してください。

設定	説明	デフォルト値
Name	**my_radius_object** など、RADIUS サーバオブジェクトの名前を一意で指定します。この設定は必須です。	デフォルト値なし
Host	RADIUS サーバのホスト名または IP アドレスを指定します。この設定は必須です。	デフォルト値なし
Service Port	RADIUS 認証トラフィックのポートを指定します。	**1812**

表 10.3 *RADIUS サーバ定義の設定*

設定	説明	デフォルト値
Secret	サーバとの間で送受信されるパケットを暗号化するまたは復号化するための秘密鍵を設定します。この設定は必須です。	デフォルト値なし
Confirm Secret	**[Secret]** 設定で指定された秘密鍵を確認します。この設定は必須です。	デフォルト値なし
Timeout	タイムアウト値を指定します。	**3**

***表 10.3** RADIUS サーバ定義の設定（続き）*

RADIUS サーバオブジェクトを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [Authentication] メニューの [RADIUS Servers] を選択します。
   [RADIUS Server List] 画面が表示されます。
3. 画面の右上で [**Create**] をクリックします。
4. **my_radius_server** など、RADIUS サーバオブジェクトの一意の名前を [**Name**] 設定に入力します。

   ***注**: 名前には小文字のみを使うようにしてください。*
5. [**Host**] 設定にリモート RADIUS サーバのホスト名または IP アドレスを入力します。
6. [**Secret**] 設定と [**Confirm Secret**] 設定に「RADIUS secret」と入力します。
7. その他の設定は必要に応じて変更します。
8. [**Finished**] をクリックします。
9. 冗長 RADIUS サーバを設定する場合は、これらの手順を繰り返して追加のサーバオブジェクトを作成します。

RADIUS 構成オブジェクトの作成

RADIUS 構成オブジェクトを作成する場合、さまざまな設定を行う必要があります。表 10.4（10-12 ページ）は、RADIUS 構成オブジェクトを構成する設定項目と値を示します。この表では、[New Authentication Configuration] 画面と同じ分類で設定項目を分けています。この構成オブジェクトを作成する詳細な手順については、「*RADIUS 構成オブジェクトを作成するには*」（10-12 ページ）を参照してください。

設定	説明	デフォルト値
Name	構成オブジェクトの名前を一意で指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **RADIUS** と指定してください。	デフォルト値なし
RADIUS Servers	BIG-IP が認証データの取得に使用する RADIUS サーバの IP アドレスを一覧表示します。 一覧表示されているサーバごとに、対応する RADIUS サーバ定義を作成する必要があります。RADIUS サーバの定義では、サーバ名、ポート番号、RADIUS 秘密鍵、およびタイムアウト値を指定します。詳細については、表 10.3（10-10 ページ）を参照してください。	デフォルト値なし
Client ID	NAS-Identifier RADIUS 属性を文字列バーと共に送信します。[**Client ID**] 設定に値を指定しないと、代わりに PAM サービスタイプが使用されます。この機能は空白のクライアント ID を指定することで無効にできます。	デフォルト値なし
Debug Logging	SYSLOG デバッグ情報を LOG_DEBUG レベルで有効にします。通常の使用ではお勧めしません。	**Disable**
Accounting Bug	アカウンティング応答ベクトルの検証を無効にします。このオプションは旧バージョンのサーバのみで必要です。	**Disable**
Retries	BIG-IP が許容する認証の再試行回数を指定します。	**3**

***表 10.4** RADIUS 構成オブジェクトの設定*

RADIUS 構成オブジェクトを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Configurations**] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [New Configuration] 画面が表示されます。
4. [**Name**] 設定に、構成オブジェクトに一意の名前（**my_radius_config** など）を入力します。
   ***注**: 名前には小文字のみを使うようにしてください。*
5. [**Type**] 設定から [**RADIUS**] を選択します。
   画面が拡張され、いくつかの設定が表示されます。
6. ここで表示されるすべての設定の値を必要に応じて変更できます。
   （高度な設定を行うには、[**Configuration**] の [**Advanced**] を選択します）
7. [**Finished**] をクリックします。

RADIUS プロファイルの作成

RADIUS 構成オブジェクトを作成した後、RADIUS プロファイルを作成または設定する必要があります。これはデフォルトの **radius** プロファイルを変更するか、デフォルトのプロファイル設定を継承したカスタムプロファイルを作成して行います。認証プロファイルの重要な機能は、既存の構成オブジェクトを参照することです。

ほとんどの場合、デフォルトのプロファイルをそのまま利用できます。ただし、デフォルトのプロファイルを使用した場合でも、作成した対応する構成オブジェクトを指定するようにデフォルトのプロファイルを変更する必要はあります。

カスタムプロファイルを作成する場合は、新しいプロファイルで継承する値を含んだ親プロファイル（カスタムプロファイルまたはデフォルトプロファイルのいずれか）を指定する必要があります。

RADIUS プロファイルを作成する場合、さまざまな設定を行う必要があります。表 10.5 は、RADIUS プロファイルを構成する設定項目と値を示します。RADIUS プロファイルの作成手順については、次の「*デフォルトの RADIUS プロファイルを変更するには*」、または「*カスタム RADIUS プロファイルを作成するには*」（10-14 ページ）を参照してください。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **RADIUS** と指定してください。	デフォルト値なし
Parent Profile	値を継承するプロファイルを指定します。	**radius**
Mode	プロファイルが有効か無効かを指定します。指定できる値は、[**Auto**]、[**Enabled**]、および [**Disabled**] です。	**Enabled**
Configuration	既存の RADIUS 構成オブジェクトを指定します。	デフォルト値なし
Rule	既存の認証ルールの名前を指定します。iRule を指定しない場合、BIG-IP は対応するデフォルト iRule を使用します。	**auth_radius**
Idle Timeout	認証または承認要求がタイムアウトになるまでの待ち時間を秒数で指定します。デフォルトの値を使用するか、別の値を指定できます。または **Indefinite** を選択することもできます。タイムアウト値の概要については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**

***表 10.5** RADIUS プロファイルの設定*

デフォルトの RADIUS プロファイルを変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Profiles**] を選択します。
   デフォルトの認証プロファイルが一覧表示されます。
3. [Name] カラムの [**radius**] をクリックします。

4. [**Mode**] 設定から [**Enabled**] または [**Auto**] を選択します。
5. [**Configuration**] 設定のリストから構成オブジェクトを選択します。[**None**] は許可されていない設定です。
6. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_radius** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_radius** を使用しない場合は、作成した既存の iRule の名前を選択してください。
7. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
8. [**Finished**] をクリックします。

カスタム RADIUS プロファイルを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [Authentication] メニューの [Profiles] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [New Profile] 画面が表示されます。
4. **my_radius_profile** など、RADIUS プロファイルの一意の名前を [**Name**] 設定に入力します。

   ***注**: 名前には小文字のみを使うようにしてください。*
5. [**Type**] 設定から [**RADIUS**] を選択します。
   画面が拡張され、追加の設定が表示されます。
6. [**Parent Profile**] 設定では、プロファイルのタイプを指定します。
   - 親プロファイルにデフォルトのプロファイル **radius** を使用する場合は、設定はそのままにします。
   - 親プロファイルにカスタムプロファイルを使用する場合は、リストからカスタムプロファイル名を選択します。
7. [**Mode**] 設定から [**Enabled**] または [**Auto**] を選択します。
8. [**Configuration**] 設定のリストから構成オブジェクトを選択します。
9. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_radius** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_radius** を使用しない場合は、作成した既存の iRule の名前を選択してください。
10. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
11. [**Finished**] をクリックします。

RADIUS サーバオブジェクトと RADIUS 構成オブジェクトを作成した後、バーチャルサーバの [**Authentication Profile**] 設定を指定して、このプロファイルをバーチャルサーバに割り当てる必要があります。

バーチャルサーバ設定を設定する方法については、第 2 章「*バーチャルサーバの設定*」を参照してください。

TACACS+ 認証モジュールの実装

TACACS+ 認証モジュールとは、BIG-IP 経由のクライアント接続を認証するための機構です。このモジュールは、リモート TACACS+ サーバに認証データが格納されている場合に使用します。この場合、クライアントの資格情報はベーシック HTTP 認証（つまりユーザ名とパスワード）に基づきます。

TACACS+ 認証モジュールを実装するには、リモート TACACS+ サーバ上のデータにアクセスするように BIG-IP を設定する必要があります。この作業は、次の手順で進めます。

- TACACS+ 構成オブジェクト
- TACACS+ プロファイル

◆ 重要

*TACACS+ オブジェクトとプロファイルが作成されると、BIG-IP はそれらを現在の管理パーティションに配置します。デフォルトプロファイルは常に **Common** パーティションに置かれます。パーティションの詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

これらのオブジェクトを作成した後、TACACS+ プロファイルをバーチャルサーバに割り当てる必要があります。

◆ 注

*TACACS+ 認証オブジェクトの作成時に、設定ユーティリティの **[Create]** ボタンをクリックできない場合は、TACACS+ 認証オブジェクトを作成する権限が現在のユーザロールに付与されていないことを示します。*

TACACS+ 構成オブジェクトの作成

TACACS+ 構成オブジェクトを作成する場合、さまざまな設定を行う必要があります。表 10.6（10-17 ページ）は、TACACS+ 構成オブジェクトを構成する設定項目と値を示します。この表では、[New Authentication Configuration] 画面と同じ分類で設定項目を分けています。このオブジェクトを作成する詳細な手順については、「*TACACS+ 構成オブジェクトを作成するには*」（10-17 ページ）を参照してください。

設定	説明	デフォルト値
Name	構成オブジェクトの名前を一意で指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **TACACS+** と指定してください。	デフォルト値なし
Servers	TACACS+ サーバのホスト名または IP アドレスを指定します。この設定は必須です。	デフォルト値なし
Secret	サーバとの間で送受信されるパケットを暗号化するまたは復号化するための秘密鍵を設定します。この設定は必須です。	デフォルト値なし
Confirm Secret	**[Secret]** 設定で指定された秘密鍵を確認します。この設定は必須です。	デフォルト値なし
Encryption	TACACS+ パケットの暗号化を有効または無効にします。通常は使用をお勧めします。	**Enabled**
Service Name	ユーザが使用認証を要求しているサービスの名前を指定します。ユーザを何の認証を要求しているかを特定することで、認証要求のタイプごとに TACACS+ サーバの動作を変えることができます。この設定は必須です。使用可能な値は、**slip**、**ppp**、**arap**、**shell**、**tty-daemon**、**connection**、**system**、**firewall** となっています。	デフォルト値なし
Protocol Name	**Service Name** 設定で指定した値に関連付けるプロトコルを指定します。このプロトコルは、クライアント認証またはシステムアカウンティングで使用されている関連サービスのサブセットとなります。使用可能な値は、**lcp**、**ip**、**ipx**、**atalk**、**vines**、**lat**、**xremote**、**tn3270**、**telnet**、**rlogin**、**pad**、**vpdn**、**ftp**、**http**、**deccp**、**osicp**、**unknown** となっています。	デフォルト値なし
Authentication	認証オプションを指定します。指定できる値は、[**Authenticate to first server**] と [**Authenticate to each server until success**] です。	**Authenticate to first server**
Accounting Information	複数の TACACS+ サーバが定義されていて、PAM セッションアカウンティングが有効な場合、アカウンティングの開始と停止のパケットを最初に使用可能なサーバまたはすべてのサーバに送信します。指定できる値は、[**Send to first available server**] と [**Send to all servers**] です。	**Send to first available server**
Debug Logging	SYSLOG デバッグ情報を LOG_DEBUG レベルで有効にします。通常の使用ではお勧めしません。	**Disable**

表 10.6 **TACACS+** *構成オブジェクトの設定*

TACACS+ 構成オブジェクトを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Configurations**] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [New Configuration] 画面が表示されます。

4. **my_tacacs_config** など、構成オブジェクトの一意の名前を [**Name**] 設定に入力します。

   ***注**: 名前には小文字のみを使うようにしてください。*

5. [**Type**] 設定から [**TACACS+**] を選択します。
   画面が拡張され、いくつかの設定が表示されます。
6. ここで表示されるすべての設定の値を必要に応じて変更できます。
   （高度な設定を行うには、[**Configuration**] の [**Advanced**] を選択します）
7. [**Finished**] をクリックします。

TACACS+ プロファイルの作成

TACACS+ 構成オブジェクトを作成した後、TACACS+ プロファイルを作成または設定する必要があります。これはデフォルトの **tacacs+** プロファイルを変更するか、デフォルトのプロファイル設定を継承したカスタムプロファイルを作成して行います。認証プロファイルの重要な機能は、既存の構成オブジェクトを参照することです。

ほとんどの場合、デフォルトのプロファイルをそのまま利用できます。ただし、デフォルトのプロファイルを使用した場合でも、作成した対応する構成オブジェクトを指定するようにデフォルトのプロファイルを変更する必要はあります。

カスタムプロファイルを作成する場合は、新しいプロファイルで継承する値を含んだ親プロファイル（カスタムプロファイルまたはデフォルトプロファイルのいずれか）を指定する必要があります。

TACACS+ プロファイルを作成する場合、さまざまな設定を行う必要があります。表 10.7（10-19 ページ）は、TACACS+ プロファイルを構成する設定項目と値を示します。TACACS+ プロファイルの作成手順については、次の「*デフォルトの TACACS+ プロファイルを変更するには*」、または「*カスタム TACACS+ プロファイルを作成するには*」（10-20 ページ）を参照してください。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **TACACS+** と指定してください。	デフォルト値なし
Parent Profile	値を継承するプロファイルを指定します。	**tacacs+**
Mode	プロファイルが有効か無効かを指定します。指定できる値は、[**Auto**]、[**Enabled**]、および [**Disabled**] です。	**Enabled**
Configuration	既存の TACACS+ 構成オブジェクトを指定します。	デフォルト値なし
Rule	既存の認証ルールの名前を指定します。iRule を指定しない場合、BIG-IP は対応するデフォルト iRule を使用します。	**auth_tacacs+**
Idle Timeout	認証または承認要求がタイムアウトになるまでの待ち時間を秒数で指定します。デフォルトの値を使用するか、別の値を指定できます。または **Indefinite** を選択することもできます。タイムアウト値の概要については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**

***表 10.7** TACACS+ プロファイルの設定*

デフォルトの TACACS+ プロファイルを変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Profiles**] を選択します。
   デフォルトの認証プロファイルが一覧表示されます。
3. Name カラムの [**tacacs+**] をクリックします。
4. [**Mode**] 設定から [**Enabled**] または [**Auto**] を選択します。
5. [**Configuration**] 設定のリストから構成オブジェクトを選択します。[**None**] は許可されていない設定です。
6. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_tacacs** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_tacacs** を使用しない場合は、作成した既存の iRule の名前を選択してください。
7. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
8. [**Finished**] をクリックします。

カスタム TACACS+ プロファイルを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [Profiles] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [New Profile] 画面が表示されます。
4. プロファイルの一意の名前を [**Name**] 設定に指定します。

   ***注**: 名前には小文字のみを使うようにしてください。*
5. [**Type**] 設定から [**TACACS+**] を選択します。
6. [**Parent Profile**] 設定では、プロファイルのタイプを指定します。
   - 親プロファイルにデフォルトのプロファイル **tacacs** を使用する場合は、設定はそのままにします。
   - 親プロファイルにカスタムプロファイルを使用する場合は、リストからカスタムプロファイル名を選択します。
7. [**Mode**] 設定から [**Enabled**] または [**Auto**] を選択します。
8. [**Configuration**] 設定のリストから構成オブジェクトを選択します。
9. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_tacacs** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_tacacs** を使用しない場合は、作成した既存の iRule の名前を選択してください。
10. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
11. [**Finished**] をクリックします。

TACACS+構成オブジェクトとTACACS+プロファイルを作成した後、バーチャルサーバの [**Authentication Profile**] 設定を指定して、このプロファイルをバーチャルサーバに割り当てる必要があります。

バーチャルサーバ設定を設定する方法については、第 2 章「*バーチャルサーバの設定*」を参照してください。

SSL クライアント証明書 LDAP 認証モジュールの実装

SSL クライアント証明書 LDAP 認証モジュールとは、BIG-IP 経由のクライアント接続を認証または承認するための機構です。SSL クライアント証明書 LDAP 認証モジュールを使用すると、リモート LDAP サーバでアプリケーショントラフィックのアクセス制御を実行できます。このモジュールは、SSL 証明書と定義済みのユーザグループおよびロールに基づいて、このアクセス制御を行います。

SSL クライアント証明書 LDAP 認証モジュールを使用すると、BIG-IP は、承認が成功または失敗したことや、LDAP サーバで何らかの資格情報が必要であることを表示できます。

また、システムは、サーバが LDAP クエリ応答で返した情報に基づいて、何らかの措置を取ることができます。たとえば、LDAP 応答情報では、ユーザのグループメンバシップや、ユーザのパスワードの期限切れなどを示すことができます。BIG-IP が LDAP 応答で特定のデータを返すように設定するには、コマンド **AUTH::subscribe**、**AUTH::unsubscribe**、および **AUTH::result-data** を使用して、iRule を作成します。詳細については、第 17 章「*iRule の記述*」の章と、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。

◆ 重要

*SSL クライアント証明書LDAP 認証モジュールを実装する前に、**Client SSL** プロファイルを設定して実装する必要があります。Client SSL プロファイルを作成する場合、**[Client Certificate]** プロパティを **[Require]** または **[Auto]** のいずれかに設定することをお勧めします。このプロパティを **[Ignore]** または **[Request]** に設定すると、SSL 接続が切断される場合があります。**Client Certificate** 設定の説明については、第 9 章「**SSL トラフィックの管理**」を参照してください。*

SSL クライアント証明書認証とは

SSL クライアント証明書 LDAP 認証では、BIG-IP は信頼できる認証局によって発行された署名入りのクライアント証明書に基づいてクライアントを承認できます。さらにグループとロールを指定することで、クライアントリクエストを承認するシステムの機能を強化できます。証明書の他にグループとロールも承認の判断基準とすることによって、システムリソースへのクライアントアクセスをフレキシブルに管理できます。

SSL クライアント証明書 LDAP モジュールを実装する前に、2 種類の資格情報について理解する必要があります。BIG-IP はこれらの資格情報を使用してリモート LDAP サーバ上のデータを使用するアプリケーショントラフィックを承認します。これらの 2 種類の資格情報は次のとおりです。

- SSL 証明書
- グループとロール

LDAP 認証に SSL 証明書を使用する

クライアントの承認中には、BIG-IP は LDAP データベースを検索する必要があります。証明書ベースの承認を利用した場合、システムは次の 3 とおりの方法で LDAP データベースを検索できます。

- **ユーザ**
  証明書が LDAP データベースに格納されていない場合は、受信したクライアントリクエストの一部として示される証明書からユーザの名前を抽出するようにシステムを設定できます。システムはユーザについてのエントリが LDAP データベースに存在するかどうかをチェックします。このシナリオは、独自の認証局を持つ会社に適しています。証明書が検証され、その後ユーザが承認されたことが確認できるためです。
- **証明書マップ**
  証明書を LDAP データベース内のユーザにマッピングするオブジェクトとクラスを作成した場合、マップ内の証明書を検索してマップからユーザを取得するようにシステムを設定できます。このときシステムは、LDAP データベース内のユーザが有効なユーザであることを確認します。
- **証明書**
  多くの LDAP サーバ環境では、既に証明書は LDAP データベースに格納されるユーザ情報に組み込まれています。LDAP サーバ環境で承認を設定する 1 つの方法は、クライアントリクエストに関連付けられたユーザに対して、受信した証明書を LDAP データベースに格納されている証明書と比較するようにシステムを設定することです。証明書がユーザの LDAP プロファイルで見つかった場合、ユーザにアクセス許可が与えられ、要求が許諾されます。

実行される証明書ベースの承認の種類と関係なく、プロセスによって表 10.8 に示すような結果が生じます。

検索結果	承認の状態
No records match	承認は失敗しています。
One record matches	承認は成功し、グループとロールが対象となります。
Two or more records match	無効なデータベースエントリによって承認は失敗しています。

***表 10.8** 検索結果と対応する承認ステータス*

LDAP 承認にグループとロールを使用

証明書ベースの承認を有効にすることに加えて、グループとロールに基づいた承認を設定することもできます。

- **グループ**
  LDAP サーバには既にグループの概念と構造が組み込まれているため、BIG-IP はその承認機能にグループを含めることができます。承認目的でグループを使用できるようにするには、システムが LDAP データベースのグループを検索する検索ベースと範囲を示す必要があります。また、グループ名とメンバ名の設定値を指定する必要があります。これらの作業を完了すると、システムは現在のユーザをメンバとしているグループが見つかるまで有効グループのリストを検索できます。BIG-IP ユーザアカウントに対するグループ全体の特権の割り当てについては、『***BIG-IP® Local Traffic Manager:Implementations***』を参照してください。

- **ロール**
  グループとは異なり、ロールはユーザに直接関連付けられた設定です。BIG-IP が実行するロールベースの承認は、すべてロールの概念が組み込まれた LDAP データベースに依存します。リソースへのアクセスをユーザに許可するべきかどうかを決定するために、BIG-IP はユーザに割り当てられたロールを検索し、そのロールを管理者によって定義された有効なロールと照合し、一致するかどうかを調べます。BIG-IP ユーザアカウントに対するグループ全体の特権の割り当てについては、『***BIG-IP® Local Traffic Manager: Implementations***』を参照してください。

SSL クライアント証明書 LDAP 認証を実装するには、リモート LDAP サーバ上のデータにアクセスするように BIG-IP を設定する必要があります。この作業は、次の手順で進めます。

- SSL クライアント証明書 LDAP 構成オブジェクト
- SSL クライアント証明書 LDAP プロファイル

◆重要

SSL Client Certificate LDAP オブジェクトとプロファイルが作成されると、BIG-IP はそれらを現在の管理パーティションに配置します。デフォルトプロファイルは常に ***Common*** *パーティションに置かれます。パーティションの詳細については、『****TMOS™ Management Guide for BIG-IP® Systems****』を参照してください。*

これらのオブジェクトを作成した後、SSL クライアント証明書 LDAP プロファイルをバーチャルサーバに割り当てる必要があります。

◆注

SSL クライアント証明書 LDAP 認証オブジェクトの作成時に、設定ユーティリティの ***[Create]*** *ボタンをクリックできない場合は、SSL クライアント証明書 LDAP 認証オブジェクトを作成する権限が現在のユーザロールに付与されていないことを示します。*

SSL クライアント証明書 LDAP 構成オブジェクトの作成

SSL クライアント証明書 LDAP 構成オブジェクトは、一連のデータと命令で構成されています。これらは対応する外部 LDAP サーバが BIG-IP から承認要求をサービスする際に必要とします。

SSL クライアント証明書 LDAP 構成オブジェクトを作成する場合、さまざまな設定を行う必要があります。表 10.9 は、この構成オブジェクトで指定できる設定の一覧とその説明です。この表では、[New Authentication Configuration] 画面と同じ分類で設定項目を分けています。このオブジェクトを設定する詳細な手順については、「*SSL クライアント証明書 LDAP 構成オブジェクトを作成するには*」(10-26 ページ）を参照してください。

設定	説明	デフォルト値
Name	構成オブジェクトの名前を一意で指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **SSL Client Certificate LDAP** と指定してください。	デフォルト値なし
Hosts	BIG-IP が認証データの取得に使用する LDAP サーバのポート番号を含む IP アドレスを一覧します。	デフォルト値なし
Search Type	システムが LDAP データベースの検索時に使用する証明書ベースの承認方法を指定します（「*LDAP 認証に SSL 証明書を使用する*」（10-22 ページ）を参照）。指定できる値は、[**User**]、[**Certificate Map**]、および [**Certificate**] です。	**User**
User Base DN	**User** と **Certificate** 検索タイプによって使用されるサブツリーの検索ベースを指定します。 一般的な検索ベースは **ou=people,dc=company,dc=com** です。この設定は必須です。詳細については、**Search Type** の設定を参照してください。	デフォルト値なし
User Key	ユーザ ID を指定する LDAP データベース内の属性名を指定します。**User** と **Certificate** の検索タイプで使用されます。よく使われる User Key 値は **uid** などです。この設定は必須です。詳細については、**Search Type** の設定を参照してください。	デフォルト値なし
Object Class	ユーザ認証方法を指定します。この設定は、検索タイプを[**Certificate**]に設定している場合にだけ表示されます。	**StrongAuthenticationUser**
Certificate Map Base DN	**Certificate Map** 検索タイプによって使用されるサブツリーの検索ベースを指定します。 一般的な検索ベースは **ou=people,dc=company,dc=com** です。この設定は必須です。詳細については、**Search Type** の設定を参照してください。	デフォルト値なし

***表 10.9** SSL クライアント証明書 LDAP 構成オブジェクトの設定*

設定	説明	デフォルト値
Certificate Map Key	ユーザ ID を指定する LDAP データベース内の属性名を指定します。**Cetificate Map** 検索タイプで使用されます。よく使われる User Key 値は **uid** などです。この設定は必須です。詳細については、**Search Type** の設定を参照してください。	デフォルト値なし
Use Serial Certificate Map	**Certificate Map** 検索タイプを選択した場合に、サブジェクトではなくクライアント証明書のシリアル番号が使用されるようにします。	**Disabled**
Cache Size	SSL セッションキャッシュに可能な最大サイズを指定します。この値を **0** に設定すると、SSL セッションのキャッシングは無効化されます。	**20000**
Cache Timeout	ネゴシエーション可能な SSL セッション ID の使用可能な継続期間を秒で指定します。この期間が過ぎると、クライアントは新しいセッションとネゴシエートする必要があります。	**300**
Secure	BIG-IP にシステムと LDAP サーバとの間の安全な通信（つまり、SSL/TLS）を使用するように指示します。	**Disabled**
Admin DN	検索を実行するためにバインドするアカウントの識別名を指定します。この*検索*アカウントは、検索を実行するために使用される読み取り専用アカウントです。**管理者**アカウントも*検索*アカウントとして利用できます。. 管理者の識別名が指定されていない場合、バインドは試行されません。サイトで匿名での検索が許可されていない場合のみ、この設定を行ってください。	デフォルト値なし
Admin Password	LDAP サーバ上に作成される*検索*アカウントのパスワードを指定します。	デフォルト値なし
Confirm Admin Password	LDAP サーバ上に作成される*検索*アカウントに指定されているパスワードを確認します。	デフォルト値なし
Group Base DN	グループ検索によって使用されるサブツリーの検索ベースを指定します。このパラメータは有効なグループを指定した場合にだけ使用されます。一般的な検索ベースは**ou=people,dc=company,dc=com**です。	デフォルト値なし
Group Key	グループサブツリー内のグループ名を指定する、LDAP データベース内の属性名を示します。	デフォルト値なし
Group Member Key	グループのメンバ（DN）を指定する、LDAP データベース内の属性名を示します。	デフォルト値なし
Valid Groups	承認されるためにユーザが所属する必要のあるグループのリストを指定します。このオプションは複数回指定できます。クライアントは、文字列で指定された 1 つのグループのメンバであれば承認されます。	デフォルト値なし
Role Key	ユーザの承認ロールを指定する、LDAP データベース内の属性名を示します。このキーは有効なロールを使用している場合にだけ使用されます（次のエントリの説明のとおり）。	デフォルト値なし

***表 10.9** SSL クライアント証明書 LDAP 構成オブジェクトの設定（続き）*

設定	説明	デフォルト値
Valid Roles	ロールのリストを指定します。ユーザは、承認されるためにこれらのロールの 1 つに割り当てられる必要があります。このリストでは、有効なロールがスペースで区切られています。このオプションは複数回指定できます。クライアントは、文字列で指定された 1 つのロールのメンバであれば承認されます。	デフォルト値なし

***表 10.9** SSL クライアント証明書LDAP 構成オブジェクトの設定（続き）*

SSL クライアント証明書 LDAP 構成オブジェクトを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Configurations**] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [**New Configuration**] 画面が表示されます。
4. 構成オブジェクトの一意の名前を [**Name**] 設定に指定します。
   ***注**: 名前には小文字のみを使うようにしてください。*
5. [**Type**] 設定から [**SSL Client Certificate LDAP**] を選択します。
6. その他のすべての設定の値を必要に応じて変更できます。（高度な設定を行うには、[**Configuration**] の [**Advanced**] を選択します）
7. [**Finished**] をクリックします。

SSL クライアント証明書 LDAP 承認プロファイルの作成

SSL クライアント証明書 LDAP 構成オブジェクトを作成した後、対応するプロファイルを作成または設定する必要があります。これはデフォルトの **ssl_cc_ldap** プロファイルを変更するか、デフォルトのプロファイル設定を継承したカスタムプロファイルを作成して行います。認証プロファイルの重要な機能は、既存の構成オブジェクトを参照することです。

ほとんどの場合、デフォルトのプロファイルをそのまま利用できます。ただし、デフォルトのプロファイルを使用した場合でも、作成した対応する構成オブジェクトを指定するようにデフォルトのプロファイルを変更する必要はあります。

カスタムプロファイルを作成する場合は、新しいプロファイルで継承する値を含んだ親プロファイル（カスタムプロファイルまたはデフォルトプロファイルのいずれか）を指定する必要があります。

SSL クライアント証明書 LDAP プロファイルを作成する場合、さまざまな設定を行う必要があります。表 10.10 は、SSL クライアント証明書 LDAP プロファイルを構成する設定項目と値を示します。このタイプのプロファイルを作成する詳細な手順については、

次の「*デフォルトの SSL クライアント証明書 LDAP プロファイルを変更するには*」、または「*カスタム SSL クライアント証明書 LDAP プロファイルを作成するには*」（10-28 ページ）を参照してください。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **LDAP (SSL Client Certificate)** と指定してください。	デフォルト値なし
Parent Profile	値を継承するプロファイルを指定します。	**ssl_cc_ldap**
Mode	プロファイルが有効か無効かを指定します。指定できる値は、[**Auto**]、[**Enabled**]、および [**Disabled**] です。	**Enabled**
Configuration	SSL クライアント証明書 LDAP 構成オブジェクトの作成	デフォルト値なし
Rule	既存の認証 iRule の名前を指定します。iRule を指定しない場合、BIG-IP は対応するデフォルト iRule を使用します。	**auth_ssl_cc_ldap**
Idle Timeout	承認要求がタイムアウトになるまでの待ち時間を秒数で指定します。デフォルトの値を使用するか、別の値を指定できます。または **Indefinite** を選択することもできます。タイムアウト値の概要については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**

***表 10.10** SSL クライアント証明書 LDAP プロファイルの設定*

デフォルトの SSL クライアント証明書 LDAP プロファイルを変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [Profiles] を選択します。
   デフォルトの認証プロファイルが一覧表示されます。
3. Name カラムの [**ssl_cc_ldap**] をクリックします。
4. [**Mode**] 設定から [**Enabled**] または [**Auto**] を選択します。
5. [**Configuration**] 設定のリストから構成オブジェクトを選択します。[**None**] は許可されていない設定です。
6. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_ssl_cc_ldap** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_ssl_cc_ldap** を使用しない場合は、作成した既存の iRule の名前を選択してください。
7. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
8. [**Finished**] をクリックします。

カスタム SSL クライアント証明書 LDAP プロファイルを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Profiles**] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [New Profile] 画面が表示されます。
4. プロファイルの一意の名前を [**Name**] 設定に指定します。

   注: 名前には小文字のみを使うようにしてください。
5. [**Type**] 設定から [**SSL Client Certificate LDAP**] を選択します。
6. [**Parent Profile**] 設定では、プロファイルのタイプを指定します。
   - 親プロファイルにデフォルトのプロファイル **ssl_cc_ldap** を使用する場合は、設定はそのままにします。
   - 親プロファイルにカスタムプロファイルを使用する場合は、リストからカスタムプロファイル名を選択します。
7. [**Mode**] 設定の画面右側にある [Custom] ボックスをクリックし、[**Enabled**] または [**Auto**] を選択します。
8. [**Configuration**] 設定の画面右側にある [Custom] ボックスをクリックしリストから構成オブジェクトを選択します。
9. [**Rule**] 設定の画面右側にある [Custom] ボックスをクリックし、認証 iRule を指定します。
   - デフォルトの iRule である **auth_ssl_cc_ldap** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_ssl_cc_ldap** を使用しない場合は、作成した既存の iRule の名前を選択してください。
10. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
11. [**Finished**] をクリックします。

SSL クライアント証明書 LDAP 構成オブジェクトと SSL クライアント証明書 LDAP プロファイルを作成した後、バーチャルサーバの [**Authentication Profile**] 設定を指定して、このプロファイルをバーチャルサーバに割り当てる必要があります。

SSL OCSP 認証モジュールの実装

SSL OCSP 認証モジュールとは、BIG-IP 経由のクライアント接続を認証するための機構です。さらに具体的に言うと、SSL OCSP 認証モジュールは、SSL 証明書の失効状態をその証明書の認証の一部としてチェックします。

Online Certificate Status Protocol (OCSP) は、公開鍵テクノロジを使用した場合に証明書失効リスト（CRL）の代替策を提供するサードパーティソフトウェアアプリケーションおよび業界標準プロトコルです。***CRL*** は、サーバシステムがクライアント証明書の検証のプロセス中にチェックできる、失効したクライアント証明書のリストです。

SSL 証明書の失効状態を確認するメカニズムとして CRL ではなく OCSP を使用する場合は、SSL OCSP 認証モジュールを実装してください。

BIG-IP は CRL と OCSP プロトコルの両方をサポートします。OCSP の代わりに CRL を使用する場合は、SSL プロファイルを設定してください。CRL の詳細は、第 9 章「*SSL トラフィックの管理*」を参照してください。

OCSP の詳細は、RFC2560（URL: ***http://www.ietf.org***）を参照してください。

OCSP の概要

OCSP を使用してクライアント証明書の失効状態を確認することは、CRL を使用するよりもかなりの利点があります。ここでは、CRL と OCSP との違い、および OCSP が動作する方法について説明します。

CRL の制限

クライアント証明書と共に表示される場合、BIG-IP では証明書を受け入れて、ターゲットサーバへの接続を転送する前に、その証明書の失効状態を調べる必要があります。失効状態を評価する標準的な方法は CRL であり、これは設定内の各マシン上の別個の CRL ファイルに格納されます。CRL は SSL 証明書の失効状態を調べる標準的な方法と考えられていますが、CRL の更新は一定の間隔ごとにのみ実施されるため、状態チェックが行われた時点ではCRL内の情報が古くなっているというリスクがあります。

また、各マシンに別々の CRL ファイルを格納させる場合、他にも次のような制限があります。

- すべての CRL ファイルは同期を保持する必要があります。
- 各マシンに別々の CRL ファイルを持つことでセキュリティ上のリスクが生じます。
- 複数の CRL ファイルを一元管理することができません。

OCSP の利点

OCSP によって、BIG-IP は証明書検証プロセス中にリアルタイムの失効状態を必ず取得します。

OCSP はクライアント/サーバモデルに基づきます。ここでクライアントシステムは証明書の失効状態を要求し、サーバシステムはその応答を送信します。このように、SSL OCSP 認証モジュールを実装すると、BIG-IP は OCSP クライアントとして動作し、OSCP レスポンダと呼ばれる外部システムは OCSP サーバとして動作します。したがって、***OCSP レスポンダ***は証明書の失効状態を要求時に BIG-IP に送信する外部サーバです。

OCSP が動作する仕組み

通常、SSL 証明書をクライアントアプリケーションから受信した後、BIG-IP（OCSP クライアントとして動作する）は、OCSP レスポンダから証明書失効状態を要求し、レスポンダから状態を受信するまで接続をブロックします。レスポンダからの状態で証明書が失効していることがわかると、BIG-IP は証明書を拒否し、接続を拒否します。レスポンダからの状態で証明書がまだ有効であることがわかると、BIG-IP は通常の証明書検証プロセスを継続し、クライアントアプリケーションを認証します。

より具体的には、アプリケーションクライアントが認証のための証明書を送信すると、BIG-IP は以下のプロセスに従います。

- まず、BIG-IP は証明書の署名者が信頼できる認証局のファイルに一覧されていることを確認します。
- 証明書がリストにある場合、BIG-IP は証明書が失効していないかどうかを確認します。OCSP なしで、CRL オプションが BIG-IP に設定されている場合、BIG-IP は証明書失効リスト（CRL）を読み取り失効状態を確認します。ただし、OCSP がある場合、BIG-IP は CRL をバイパスし、適切な OCSP レスポンダに失効状態の要求を送信する準備をします。
- 次に BIG-IP は、最初のレスポンダがクライアント証明書に署名した認証局と一致していない場合でも、そのレスポンダにクエリを送信します。ただし、接続エラーによって最初のレスポンダとの接続に失敗した場合は、次のレスポンダにクエリが送信されます。
- 次に BIG-IP は認証局を SSL OCSP プロファイルに一覧されている CA と照合し、一致するかどうか調べます。
- 一致する場合、BIG-IP はクライアント証明書の [**AuthorityInfoAccess**（**AIA**）] フィールド（フィールドが存在する場合）内の対象 URL を確認し、その URL を使用して証明書失効状態の要求を OCSP レスポンダに送信します。
- [**Ignore AIA**] パラメータが SSL OCSP プロファイル内で有効化されている場合、BIG-IP は代わりに一致する SSL OCSP プロファイルの **url** パラメータに指定されている URL を使用して、証明書失効状態の要求を送信します。

- 一致がない場合、BIG-IP はシステムに定義されている別の SSL OCSP プロファイルの **calist** 設定を確認します。すべての SSL OCSP プロファイルが確認され、一致が見つからない場合、証明書の検証は失敗し、BIG-IP は元のクライアントリクエストを拒否します。
- BIG-IP はレスポンダから証明書失効状態を受け取ると、元のクライアントリクエストに証明書状態ヘッダを挿入します（BIG-IP がそのように設定されている場合）。証明書状態ヘッダの名前は、**SSLClientCertificateStatus** です。ヘッダの詳細は、次の「*BIG-IP プロファイルの前提条件*」を参照してください。

SSL OCSP 認証モジュールを実装するには、リモート OCSP サーバ上のデータにアクセスするように BIG-IP を設定する必要があります。この作業は、次の手順で進めます。

- SSL OCSP レスポンダオブジェクトを作成する
- SSL OCSP 構成オブジェクトを作成する
- SSL OCSP プロファイルを作成する

◆ 重要

*SSL OCSP オブジェクトとプロファイルが作成されると、BIG-IP はそれらを現在の管理パーティションに配置します。デフォルトプロファイルは常に **Common** パーティションに置かれます。パーティションの詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

これらのオブジェクトを作成した後、OCSP プロファイルをバーチャルサーバに割り当てる必要があります。

1 つの SSL OCSP プロファイルは特定のレスポンダをターゲットとすることができ、複数の SSL OCSP プロファイルは同じレスポンダをターゲットとすることができます。各レスポンダ自身は認証局（CA）に関連付けられ、複数のレスポンダに同じ CA を関連付けることができます。

◆ 注

BIG-IP では、CRL オプションと OCSP オプションの両方を有効にできます。ほんとんどのユーザは両方を有効する必要はなく、どちらか一方を有効にすればすみます。ただし、両方のオプションを有効にする必要のあるまれなケースでは、CRL ファイルの検索とレスポンダへの接続の両方がうまくいく必要があります。そうでない場合、BIG-IP は状態を取得できません。

BIG-IP プロファイルの前提条件

SSL OCSP 認証モジュールの設定では、OCSP レスポンダオブジェクトと SSL OCSP プロファイルだけでなく、他に HTTP と Client SSL という 2 つのプロファイルも作成する必要があります。

Client SSL プロファイルを作成する場合、証明書を要求するプロファイルを設定することをお勧めします。これによって、[**SSLClientCertificateStatus**] ヘッダの BIG-IP の使用が最適化されます。

BIG-IP はクライアントリクエストにヘッダを挿入するように（HTTP プロファイルまたは iRule を介して）事前設定されている場合のみこのヘッダを挿入します。

次のセクションでは、OCSP レスポンダオブジェクト、SSL OCSP 構成オブジェクト、および SSL OCSP プロファイルを作成する方法について説明します。

◆注

*SSL OCSP 認証オブジェクトの作成時に、設定ユーティリティの **[Create]** ボタンをクリックできない場合は、SSL OCSP 認証オブジェクトを作成する権限が現在のユーザロールに付与されていないことを示します。*

OCSP レスポンダオブジェクトの作成

OCSP レスポンダオブジェクトとは、外部 OCSP レスポンダの URL を含めるように作成されるオブジェクトです。外部 OCSP レスポンダごとに別個の OCSP レスポンダオブジェクトを作成する必要があります。

続けて OCSP 構成オブジェクトを作成すると、構成オブジェクトには作成した OCSP レスポンダオブジェクトへの参照が含まれます。

OCSP レスポンダオブジェクトを作成する場合、いくつかの設定を行う必要があります。表 10.11 は、SSL OCSP レスポンダオブジェクトを構成する設定項目と値を示します。このオブジェクトを作成する詳細な手順については、「*OCSP レスポンダオブジェクトを作成するには*」（10-34 ページ）を参照してください。

設定	説明	デフォルト値
Name	構成オブジェクトの名前を一意で指定します。この設定は必須です。	デフォルト値なし
URL	レスポンダ上の OCSP サービスへの問い合わせで使用される URL を指定します。	デフォルト値なし
Certificate Authority File	OCSP 応答上の署名を検証するために使用される、信頼できる認証局の証明書を含むファイルの名前を指定します。	デフォルト値なし
Certificate Authority Path	OCSP 応答上の署名を検証するために使用される、信頼できる認証局の証明書を含むパスの名前を指定します。	デフォルト値なし
Verify Other	応答から証明書が除外されている場合に、証明書に署名する OCSP 応答を検索するために使用されるファイルの名前を指定します。	デフォルト値なし
Trust Other	**Verify Other** 設定で指定される証明書を信頼するように BIG-IP に指示します。	**Disable**
VA File	明示的に信頼されたレスポンダの証明書を含むファイルの名前を指定します。このパラメータは、レスポンダの認証局ストアに既にロードされた証明書によってレスポンダがカバーされていない場合に必要です。	デフォルト値なし

***表 10.11** OCSP レスポンダの設定*

設定	説明	デフォルト値
Signer	OCSP 要求を署名するために使用される証明書を指定します。 特殊な意味 : 証明書が指定されているがキーが指定されていない場合、秘密鍵は証明書と同じファイルから読み取られます。 証明書とキーのどちらも指定されていない場合、要求は署名されません。 証明書は指定されていないがキーは指定されている場合、設定は無効とみなされます。	デフォルト値なし
Signing Key	OCSP 要求に署名するために使用されるキーを指定します。	デフォルト値なし
Sign Other	OCSP 要求に追加する追加証明書を一覧します。	デフォルト値なし
Sign Digest	署名証明書とキーを使用して要求に署名するためのアルゴリズムを指定します。 特殊な意味 : 署名要求が有効でない（つまり **request signing certificate** パラメータと **request signing key** パラメータの両方が空）場合、このパラメータは意味がありません。 このパラメータは署名要求が有効な場合にのみ必要です。	**Sha1**
Validity Period	BIG-IP が受諾できるエラー範囲を指定するために使用する秒数を指定します。この設定は、OCSP レスポンダクロックとクライアントクロックが同期されていない場合に使用されます。同期されていないと証明書状態の確認が失敗となることがあります。この値には正の数を指定してください。このパラメータは必須項目です。	**300**
Status Age	状態応答の **notBefore** フィールドと比較するための時間を、秒数で指定します。状態応答が [**notAfter**] フィールドを含まない場合に使用されます。	**0**
Ignore AIA	証明書の [**AIA**] フィールドに含まれる URL を無視して、常にレスポンダが指定する URL を代わりに使用するように BIG-IP に指示します。	Disabled
Trust Other	証明書を明示的に信頼すること、証明書に対してその他のチェックは行わないことを指定します。	Disabled
Allow Certificates	OCSP 要求への証明書の追加を許可します。	Enabled
Verify	BIG-IP が OCSP 応答署名または臨時の値を検証します。デバッグのみに使用されます。	Enabled
Intern	BIG-IP が署名者の証明書を検索する際に、OCSP 応答内に含まれる証明書を無視します。この設定を使用するには、署名者の証明書を [**Verify Other**] または [**VA File**] 設定のいずれかに指定する必要があります。	Enabled
Verify Signature	BIG-IP が OCSP 応答上の署名を確認します。テスト目的でのみ使用されます。	Enabled
Verify Certificate	BIG-IP が OCSP 応答上の証明書を検証します。	Enabled
Certificate Chain	BIG-IP が OCSP 応答の証明書からのチェーンを構築します。	Enabled
Check Certificates	BIG-IP は、署名者の証明書が必要な状態情報を提供するよう承認されているかどうかを確認する追加のチェックを実行します。テスト目的でのみ使用されます。	Enabled

***表 10.11** OCSP レスポンダの設定（続き）*

設定	説明	デフォルト値
Explicit OCSP	BIG-IP は OCSP 応答の署名者の証明書が OCSP 応答署名について承認されていることを明示的に信頼します。署名者の証明書が **OCSP signing** エクステンションを含まない場合、この設定の指定によって応答は信頼されません。	Enabled

***表 10.11** OCSP レスポンダの設定（続き）*

OCSP レスポンダオブジェクトを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
2. [Authentication] メニューの [OCSP Responders] を選択します。
3. 画面の右上で [**Create**] をクリックします。
4. **my_ocsp_responder** など、レスポンダオブジェクトの一意の名前を [**Name**] 設定に入力します。

   ***注**: 名前には小文字のみを使うようにしてください。*
5. [**URL**] 設定に外部レスポンダの URL を入力します。
6. その他のすべての設定の値を必要に応じて変更できます。（高度な設定を行うには、[**Configuration**] ヘッダの [**Advanced**] を選択します）。
7. [**Finished**] をクリックします。

SSL OCSP 構成オブジェクトの作成

SSL OCSP 構成オブジェクトは、1 つ以上の OCSP レスポンダオブジェクトを参照するオブジェクトです。

OCSP 構成オブジェクトを作成する場合、さまざまな設定を行う必要があります。表 10.12 は、SSL OCSP 構成オブジェクトで指定できる設定の一覧とその説明です。このオブジェクトを設定する詳細な手順については、次の「*SSL OCSP 構成オブジェクトを作成するには*」を参照してください。

設定	説明	デフォルト値
Name	構成オブジェクトの名前を一意で指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **SSL OCSP** と指定してください。	デフォルト値なし
Responders	SSL証明書の失効状態の確認に使用するOCSPレスポンダを指定します。	デフォルト値なし

***表 10.12** SSL OCSP 構成オブジェクトの設定*

SSL OCSP 構成オブジェクトを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Configurations**] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [New Configuration] 画面が表示されます。
4. [**Name**] 設定に、構成オブジェクトに一意の名前（**my_ocsp_config** など）を入力します。
   注: 名前には小文字のみを使うようにしてください。
5. [**Type**] 設定から [**SSL OCSP**] を選択します。
6. [**Responders**] 設定にすべてのレスポンダオブジェクトを指定します。
7. [**Finished**] をクリックします。

SSL OCSP プロファイルの作成

SSL OCSP 構成オブジェクトを作成した後、SSL OCSP プロファイルを作成または設定する必要があります。これはデフォルトの **ssl_ocsp** プロファイルを変更するか、デフォルトのプロファイル設定を継承したカスタムプロファイルを作成して行います。認証プロファイルの重要な機能は、既存の構成オブジェクトを参照することです。

ほとんどの場合、デフォルトのプロファイルをそのまま利用できます。ただし、デフォルトのプロファイルを使用した場合でも、作成した対応する構成オブジェクトを指定するようにデフォルトのプロファイルを変更する必要はあります。

カスタムプロファイルを作成する場合は、新しいプロファイルで継承する値を含んだ親プロファイル（カスタムプロファイルまたはデフォルトプロファイルのいずれか）を指定する必要があります。

OCSP プロファイルを作成する場合、さまざまな設定を行う必要があります。表 10.13 は、SSL OCSP プロファイルを構成する設定項目と値を示します。OCSP プロファイルの作成手順については、「*デフォルトの SSL OCSP プロファイルを変更するには*」（10-36 ページ）または「*カスタム SSL OCSP プロファイルを作成するには*」（10-36 ページ）を参照してください。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **OCSP** と指定してください。	デフォルト値なし
Parent Profile	値を継承するプロファイルを指定します。	**ssl_ocsp**

***表 10.13** SSL OCSP プロファイルの設定*

設定	説明	デフォルト値
Mode	プロファイルが有効か無効かを指定します。指定できる値は、[**Auto**]、[**Enabled**]、および [**Disabled**] です。	**Enabled**
Configuration	既存の OCSP 構成オブジェクトの名前を指定します。この設定は必須です。	デフォルト値なし
Rule	既存の認証 iRule の名前を指定します。iRule を指定しない場合、BIG-IP は対応するデフォルト iRule を使用します。	**auth_ssl_ocsp**
Idle Timeout	認証要求がタイムアウトになるまでの待ち時間を秒数で指定します。デフォルトの値を使用するか、別の値を指定できます。または **Indefinite** を選択することもできます。タイムアウト値の概要については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**

***表 10.13** SSL OCSP プロファイルの設定（続き）*

デフォルトの SSL OCSP プロファイルを変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Profiles**] を選択します。
   デフォルトの認証プロファイルが一覧表示されます。
3. [Name] カラムの [**ssl_ocsp**] をクリックします。
4. [**Mode**] 設定から [**Enabled**] または [**Auto**] を選択します。
5. [**Configuration**] 設定のリストから構成オブジェクトを選択します。[**None**] は許可されていない設定です。
6. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_ocsp** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_ocsp** を使用しない場合は、作成した既存の iRule の名前を選択してください。
7. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
8. [**Finished**] をクリックします。

カスタム SSL OCSP プロファイルを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Profiles**] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [New Profile] 画面が表示されます。
4. プロファイルの一意の名前を [**Name**] 設定に指定します。
   ***注**: 名前には小文字のみを使うようにしてください。*

5. [**Type**] 設定から [**SSL OCSP**] を選択します。
6. [**Parent Profile**] 設定では、プロファイルのタイプを指定します。
   - 親プロファイルにデフォルトのプロファイル **ssl_ocsp** を使用する場合は、設定はそのままにします。
   - 親プロファイルにカスタムプロファイルを使用する場合は、リストからカスタムプロファイル名を選択します。
7. [**Mode**] 設定の画面右側にある [Custom] ボックスをクリックし、[**Enabled**] または [**Auto**] を選択します。
8. [**Configuration**] 設定の画面右側にある [Custom] ボックスをクリックしリストから構成オブジェクトを選択します。
9. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_ocsp** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_ocsp** を使用しない場合は、作成した既存の iRule の名前を選択してください。
10. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
11. [**Finished**] をクリックします。

SSL OCSP レスポンダオブジェクト、SSL OCSP 構成オブジェクト、および SSL OCSP プロファイルを作成した後、バーチャルサーバの [**Authentication Profile**] 設定を指定して、このプロファイルをバーチャルサーバに割り当てる必要があります。

バーチャルサーバ設定を設定する方法については、第 2 章「*バーチャルサーバの設定*」を参照してください。

CRLDP 認証モジュールの実装

CRLDP 認証モジュールとは、BIG-IP 経由のクライアント接続を認証するための機構です。CRLDP 認証機能は、「***CRLDP (Certificate Revocation List Distribution Points)***」というテクノロジに基づいています。CRLDP は業界標準のプロトコルであり、標準の証明書失効リスト（CRL）をチェックして失効状態を特定する代替手段を提供するほか、OCSP（Online Certificate Status Protocol）の代わりに使用することもできます。OCSP については「*SSL OCSP 認証モジュールの実装*」（10-29 ページ）を参照してください。

CRLDP 認証モジュールは、証明書認証の一環として、CRL 配布点を利用して SSL 証明書の失効状態をチェックします。***CRL 配布点***は、ネットワーク全域に証明書失効情報を配布するためのメカニズムです。具体的に言えば、証明書で指定された URI（Uniform Resource Identifier）もしくはディレクトリ名のことであり、これらはサーバが CRL 情報を取得する方法を示しています。配布点と CRL を組み合わせて使うことで、任意の数の LDAP サーバを使用する認証局を設定することができます。

BIG-IP で、SSL 証明書の失効状態を確認するメカニズムとして CRL 配布点を使用したい場合は、CRLDP 認証モジュールを実装します。

CRLDP 認証モジュールを実装するには、リモート CRLDP サーバ上のデータにアクセスするように BIG-IP を設定する必要があります。この作業は、次の手順で進めます。

- 1 つ以上の上位 CRLDP サーバオブジェクトを作成する
- CRLDP 構成オブジェクトを作成する
- CRLDP プロファイルを作成する

◆ 重要

*CRLDP オブジェクトとプロファイルが作成されると、BIG-IP はそれらを現在の管理パーティションに配置します。デフォルトプロファイルは常に **Common** パーティションに置かれます。パーティションの詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

これらのオブジェクトを作成した後、CRLDP プロファイルをバーチャルサーバに割り当てる必要があります。

◆ 注

*CRLDP 認証オブジェクトの作成時に、設定ユーティリティの **[Create]** ボタンをクリックできない場合は、CRLDP 認証オブジェクトを作成する権限が現在のユーザロールに付与されていないことを示します。*

前提条件

CRLDP 認証モジュールの設定では、Client SSL プロファイルを前もって作成しておく必要があります。Client SSL プロファイルを作成する場合、[**Client Certificate**] を [**Request**] または [**Require**] のいずれかに設定することをお勧めします。Client SSL プロファイルの作成については、第 9 章「*SSL トラフィックの管理*」を参照してください。

Client SSL プロファイルを作成し、[**Client Certificate**] を正しく設定すれば、CRLDP サーバオブジェクト、CRLDP 構成オブジェクト、CRLDP プロファイルを作成することができます。

CRLDP サーバオブジェクトの作成

CRLDP サーバオブジェクトを作成する場合、いくつかの設定を行う必要があります。表 10.14 は、CRLDP サーバオブジェクトを構成する設定項目と値を示します。サーバオブジェクトを作成する詳細な手順については、次の「*CRLDP サーバオブジェクトを作成するには*」を参照してください。

設定	説明	デフォルト値
Name	**my_crldp_server** など、CRLDP サーバオブジェクトの名前を一意で指定します。この設定は必須です。	デフォルト値なし
Address	CRLDP サーバの IP アドレスを指定します。この設定は必須です。	デフォルト値なし
Service Port	CRLDP 認証トラフィックのポートを指定します。	**389**
Base DN	CRL 配布点をディレクトリ名 (**dirName**) 形式で指定する、証明書の LDAP ベース識別名を指定します。X509v3 属性 **crlDistributionPoints** の値が **dirName** 形式の場合に使用します。この場合、BIG-IP は **crlDistributionPoints 属性**の値と **Base DN** 値の照合を試みます。**Base DN** 値は **cn=lxxx,dc=f5,dc=com** のように指定します。	デフォルト値なし
Reverse DN	Base DN 値と X509v3 属性 **crlDistributionPoints** の値との照合の順番を指定します。可能な値は、[**Enabled**] と [**Disabled**] です。 [**Enabled**] に設定すると、証明書の **dirName** 文字列 (**c=us,st=wa,l=sea,ou=f5,cn=xxx** など) に応じて、左から右、もしくは DN 文字列の先頭から順に、ベース DN の照合が行われます。	**Disabled**

***表 10.14** CRLDP サーバ定義の設定*

CRLDP サーバオブジェクトを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [Authentication] メニューの [CRLDP Servers] を選択します。
   [CRLDP Server List] 画面が表示されます。

3. 画面の右上で [**Create**] をクリックします。
4. **my_crldp_server** など、CRLDP サーバオブジェクトの一意の名前を [**Name**] 設定に入力します。

   注: 名前には小文字のみを使うようにしてください。
5. [**Address**] 設定で、リモート CRLDP サーバの IP アドレスを入力します。
6. [**Base DN**] 設定で、ベース識別名を入力します。
7. [**Reverse DN**] リストで、デフォルト値をそのまま使用するか、[**Enabled**] を選択します。
8. [**Finished**] をクリックします。
9. 冗長 CRLDP サーバを設定する場合は、これらの手順を繰り返して追加のサーバオブジェクトを作成します。

CRLDP 構成オブジェクトの作成

CRLDP 構成オブジェクトを作成する場合、さまざまな設定を行う必要があります。表 10.15 は、CRLDP 構成オブジェクトを構成する設定項目と値を示します。この表では、[New Authentication Configuration] 画面と同じ分類で設定項目を分けています。この構成オブジェクトを作成する詳細な手順については、「*CRLDP 構成オブジェクトを作成するには*」（10-41 ページ）を参照してください。

設定	説明	デフォルト値
Name	構成オブジェクトの名前を一意で指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **CRLDP** と指定してください。	デフォルト値なし
Connection Timeout	接続がタイムアウトになるまでの秒数を指定します。	**15**
Update Interval	CRL 配布点の更新インターバルを指定します。CRL 配布点の更新インターバルを設定すると、CRL タイムアウト値に関係なく、CRL の状態を定期的にチェックすることができます。これによって、BIG-IP が証明書の状態をチェックする前に CRL 情報が古くなってしまうということがなくなります。	**0**
Use Issuer	そのクライアント証明書発行者の証明書から CRL 配布点を抽出すべきかどうかを示します。	**Disable**
CRLDP Servers	BIG-IP が認証データの取得に使用する CRLDP サーバの IP アドレスを一覧表示します。 一覧表示されているサーバごとに、対応する CRLDP サーバ定義を作成する必要があります。CRLDP サーバの定義では、サーバ名、IP アドレス、ポート番号、ベース DN、リバース DN を指定します。詳細については、「*CRLDP サーバオブジェクトを作成するには*」（10-39 ページ）を参照してください。	デフォルト値なし

***表 10.15** CRLDP 構成オブジェクトの設定*

CRLDP 構成オブジェクトを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Configurations**] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [**New Configuration**] 画面が表示されます。
4. [**Name**] 設定に、構成オブジェクトに一意の名前（**my_crldp_config** など）を入力します。
   注: 名前には小文字のみを使うようにしてください。
5. [**Type**] 設定から [**CRLDP**] を選択します。
   画面が拡張され、いくつかの設定が表示されます。
6. ここで表示されるすべての設定の値を必要に応じて変更できます。
7. [**Finished**] をクリックします。

CRLDP プロファイルの作成

CRLDP 構成オブジェクトを作成した後、CRLDP プロファイルを作成または設定する必要があります。これはデフォルトの **ssl_crldp** プロファイルを変更するか、デフォルトのプロファイル設定を継承したカスタムプロファイルを作成して行います。認証プロファイルの重要な機能は、既存の構成オブジェクトを参照することです。

ほとんどの場合、デフォルトのプロファイルをそのまま利用できます。ただし、デフォルトのプロファイルを使用した場合でも、作成した対応する構成オブジェクトを指定するようにデフォルトのプロファイルを変更する必要はあります。

カスタムプロファイルを作成する場合は、新しいプロファイルで継承する値を含んだ親プロファイル（カスタムプロファイルまたはデフォルトプロファイルのいずれか）を指定する必要があります。

CRLDP プロファイルを作成する場合、さまざまな設定を行う必要があります。表 10.16 は、CRLDP プロファイルを構成する設定項目と値を示します。CRLDP プロファイルを作成する詳細な手順については、「*カスタム CRLDP プロファイルを作成するには*」（10-42 ページ）を参照してください。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **CRLDP** と指定してください。	デフォルト値なし
Parent Profile	値を継承するプロファイルを指定します。	**ssl_crldp**
Mode	プロファイルが有効か無効かを指定します。可能な設定は、[**Enabled**] および [**Disabled**] です。	**Enabled**

***表 10.16** CRLDP プロファイルの設定*

設定	説明	デフォルト値
Configuration	既存の CRLDP 構成オブジェクトを指定します。	デフォルト値なし
Rule	既存の認証ルールの名前を指定します。iRule を指定しない場合、BIG-IP は対応するデフォルト iRule を使用します。	**auth_ssl_crldp**
Idle Timeout	認証または承認要求がタイムアウトになるまでの待ち時間を秒数で指定します。デフォルトの値を使用するか、別の値を指定できます。または **Indefinite** を選択することもできます。タイムアウト値の概要については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**

***表 10.16** CRLDP プロファイルの設定（続き）*

デフォルト CRLDP プロファイルを変更するには

1. ナビゲーションペインの [Main] タブで [**Local Traffic**] を展開し、[**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Profiles**] を選択します。
   デフォルトの認証プロファイルが一覧表示されます。
3. Name カラムの [**ssl_crldp**] をクリックします。
4. [**Mode**] 設定で、モードが [**Enabled**] に設定されていることを確認します。
5. [**Configuration**] 設定の画面右側にある Custom ボックスをクリックしリストから構成オブジェクトを選択します。[**None**] は許可されていない設定です。
6. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_ssl_crldp** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_ssl_crldp** を使用しない場合は、画面右の [Custom] ボックスをクリックして、前に作成した既存の iRule の名前を選択します。
7. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
8. [**Finished**] をクリックします。

カスタム CRLDP プロファイルを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [Authentication] メニューの [Profiles] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [New Profile] 画面が表示されます。
4. [**Name**] 設定で、CRLDP プロファイルに一意の名前（**my_crldp_profile** など）を入力します。

注: 名前には小文字のみを使うようにしてください。

5. [**Type**] 設定から [**CRLDP**] を選択します。
画面が拡張され、追加の設定が表示されます。
6. [**Parent Profile**] 設定では、プロファイルのタイプを指定します。
   - 親プロファイルにデフォルトのプロファイル **ssl_crldp** を使用する場合は、設定はそのままにします。
   - 親プロファイルにカスタムプロファイルを使用する場合は、リストからカスタムプロファイル名を選択します。
7. [**Mode**] 設定で、モードが [**Enabled**] に設定されていることを確認します。
8. [**Configuration**] 設定の画面右側にある Custom ボックスをクリックしリストから構成オブジェクトを選択します。
9. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_ssl_crldp** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_ssl_crldp** を使用しない場合は、画面右の [Custom] ボックスをクリックして、前に作成した既存の iRule の名前を選択します。
10. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
11. [**Finished**] をクリックします。

CRLDP サーバオブジェクトと CRLDP 構成オブジェクトを作成した後、バーチャルサーバの [**Authentication Profile**] 設定を指定して、このプロファイルをバーチャルサーバに割り当てる必要があります。

バーチャルサーバ設定を設定する方法については、第 2 章「*バーチャルサーバの設定*」を参照してください。

Kerberos 委任認証モジュールの実装

Kerberos 委任モジュールとは、BIG-IP 経由のクライアント接続を認証するための機構です。Microsoft® Windows® Integrated Authentication を使ってアプリケーショントラフィックの認証を行っている場合は、このモジュールを使用することができます。

◆ 重要

*Kerberos 委任モジュールを実装する際には、前もって BIG-IP に DNS（Domain Name Server）サーバを接続して、BIG-IP をドメインに追加しておく必要があります。詳細については、『**BIG-IP® Local Traffic Manager: Implementations**』を参照してください。*

Kerberos 委任モジュールは、クライアントプリンシパルの委任 Kerberos クレデンシャルを取得し、続いてサーバ側のプリンシパルの Kerberos クレデンシャルを取り出します。Kerberos 委任モジュールは、基本的に Kerberos クレデンシャルのプロキシとして動作します。つまり、ドメイン内のサーバに接続すると、ブラウザクライアントが Kerberos クレデンシャルをフェッチします。「***委任クレデンシャル***」と呼ばれるこれらのクレデンシャルは BIG-IP に渡され、次に BIG-IP がバックエンドにある実際のサーバのクレデンシャルを取り出して、それらをクライアントに返します。

Kerberos 委任認証モジュールを実装するには、リモートサーバ上のデータにアクセスするように BIG-IP を設定する必要があります。この作業は、次の手順で進めます。

- Kerberos 委任構成オブジェクト
- Kerberos 委任プロファイル

◆ 重要

*Kerberos 委任オブジェクトとプロファイルが作成されると、BIG-IP はそれらを現在の管理パーティションに配置します。デフォルトプロファイルは常に **Common** パーティションに置かれます。パーティションの詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

これらのオブジェクトを作成した後、Kerberos 委任プロファイルをバーチャルサーバに割り当てる必要があります。

◆ 注

*Kerberos 委任認証オブジェクトの作成時に、設定ユーティリティの **[Create]** ボタンをクリックできない場合は、Kerberos 委任認証オブジェクトを作成する権限が現在のユーザロールに付与されていないということになります。*

Kerberos 委任構成オブジェクトの作成

Kerberos 委任構成オブジェクトを作成する場合、さまざまな設定を行う必要があります。表 10.17 は、Kerberos 委任構成オブジェクトを構成する設定項目と値を示します。この表では、[New Authentication Configuration] 画面と同じ分類で設定項目を分けています。このオブジェクトを作成する詳細な手順については、*Kerberos 委任設定オブジェクトを作成するには*を参照してください。

設定	説明	デフォルト値
Name	構成オブジェクトの名前を一意で指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **Kerberos Delegation** と指定してください。	デフォルト値なし
Client Principal Name	**HTTP/<fqdn>** というフォーマットを使って、クライアントプリンシパルの名前を指定します。**fqdn** には、作成するバーチャルサーバの完全修飾ドメイン名が入ります。詳細については、『***BIG-IP® Local Traffic Manager: Implementations***』を参照してください。	デフォルト値なし
Server Principal Name	**HTTP/<fqdn>** というフォーマットを使って、バックエンド Web サーバのプリンシパルの名前を指定します。**fqdn** には、プール内の Web サーバの完全修飾ドメイン名が入ります。詳細については、『***BIG-IP® Local Traffic Manager: Implementations***』を参照してください。	デフォルト値なし
Debug Logging	SYSLOG デバッグ情報を LOG_DEBUG レベルで有効にします。通常の使用ではお勧めしません。	**Disable**

表 10.17 *Kerberos 委任構成オブジェクトの設定*

Kerberos 委任設定オブジェクトを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Configurations**] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [New Configuration] 画面が表示されます。
4. [**Name**] 設定に、構成オブジェクトに一意の名前（**my_kerberos_config** など）を入力します。
   注: 名前には小文字のみを使うようにしてください。
5. [**Type**] 設定から [**Kerberos Delegation**] を選択します。
   画面が拡張され、いくつかの設定が表示されます。
6. ここで表示されるすべての設定の値を必要に応じて変更できます。
   （高度な設定を行うには、[**Configuration**] の [**Advanced**] を選択します）
7. [**Finished**] をクリックします。

Kerberos 委任プロファイルの作成

Kerberos 委任構成オブジェクトを作成したら、次に Kerberos 委任プロファイルの作成または設定を行う必要があります。これはデフォルトの **krbdelegate** プロファイルを変更するか、デフォルトのプロファイル設定を継承したカスタムプロファイルを作成して行います。認証プロファイルの重要な機能は、既存の構成オブジェクトを参照することです。

ほとんどの場合、デフォルトのプロファイルをそのまま利用できます。ただし、デフォルトのプロファイルを使用した場合でも、作成した対応する構成オブジェクトを指定するようにデフォルトのプロファイルを変更する必要はあります。

カスタムプロファイルを作成する場合は、新しいプロファイルで継承する値を含んだ親プロファイル（カスタムプロファイルまたはデフォルトプロファイルのいずれか）を指定する必要があります。

Kerberos Delegation プロファイルを作成する場合、さまざまな設定を行う必要があります。表 10.18 は、Kerberos Delegation プロファイルを構成する設定項目と値を示します。Kerberos Delegation プロファイルの作成手順については、次の*デフォルトの Kerberos 委任プロファイルを変更するには*、または「*カスタム Kerberos Delegation プロファイルを作成するには*」（10-47 ページ）を参照してください。

設定	説明	デフォルト値
Name	プロファイルの一意の名前を指定します。この設定は必須です。	デフォルト値なし
Type	実装する認証モジュールの種別を指定します。この値は **Kerberos Delegation** と指定してください。	デフォルト値なし
Parent Profile	値を継承するプロファイルを指定します。	**krbdelegate**
Mode	プロファイルが有効か無効かを指定します。指定できる値は、[**Auto**]、[**Enabled**]、および [**Disabled**] です。	**Enabled**
Configuration	既存の Kerberos Delegation 構成オブジェクトを指定します。	デフォルト値なし
Rule	既存の認証ルールの名前を指定します。iRule を指定しない場合、BIG-IP は対応するデフォルト iRule を使用します。	**auth_krbdelegate**
Idle Timeout	認証または承認要求がタイムアウトになるまでの待ち時間を秒数で指定します。デフォルトの値を使用するか、別の値を指定できます。または **Indefinite** を選択することもできます。タイムアウト値の概要については、第 1 章「*ローカルトラフィック管理の概要*」を参照してください。	**300**
Cookie Name	このプロファイルに関連付ける Cookie の名前を一意で指定します。	**f5auth**
Cookie Key	Cookie データの暗号化に使用する暗号化キーを、強力なパスワード形式で指定します。デフォルト値を知っている攻撃者が Cookie データを復号化し、信頼できるユーザになりすますことがないように、F5 ではデフォルト値を変更することを推奨しています。	**abc123**

***表 10.18** Kerberos 委任プロファイルの設定*

デフォルトの Kerberos 委任プロファイルを変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [**Profiles**] を選択します。
   デフォルトの認証プロファイルが一覧表示されます。
3. [Name] カラムの [**krbdelegate**] をクリックします。
4. [**Mode**] 設定では、デフォルト値の [**Enabled**] をそのまま使用します。
5. [**Configuration**] 設定のリストから構成オブジェクトを選択します。[**None**] は許可されていない設定です。
6. [**Rule**] 設定で認証 iRule を指定します。
   - デフォルトの iRule である **auth_krbdelegate** を使用する場合は、この設定はこのままにしておいてください。
   - デフォルトの iRule である **auth_krbdelegate** を使用しない場合は、作成した既存の iRule の名前を選択してください。
7. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
8. [**Finished**] をクリックします。

カスタム Kerberos Delegation プロファイルを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Profiles**] をクリックします。
   [Profiles] 画面が開きます。
2. [**Authentication**] メニューの [Profiles] を選択します。
3. 画面の右上で [**Create**] をクリックします。
   [New Profile] 画面が表示されます。
4. プロファイルの一意の名前を [**Name**] 設定に指定します。

   注: 名前には小文字のみを使うようにしてください。
5. [**Type**] 設定から [**Kerberos Delegation**] を選択します。
6. [**Parent Profile**] 設定では、プロファイルのタイプを指定します。
   - 親プロファイルにデフォルトのプロファイル **krbdelegate** を使用する場合は、設定はそのままにします。
   - 親プロファイルにカスタムプロファイルを使用する場合は、リストからカスタムプロファイル名を選択します。
7. [**Mode**] 設定では、デフォルト値の [**Enabled**] をそのまま使用します。
8. [**Configuration**] 設定のリストから構成オブジェクトを選択します。
9. [**Rule**] 設定で認証 iRule を指定します。

- デフォルトの iRule である **auth_krbdelegate** を使用する場合は、この設定はこのままにしておいてください。
- デフォルトの iRule である **auth_krbdelegate** を使用しない場合は、作成した既存の iRule の名前を選択してください。

10. [**Idle Timeout**] 設定では、デフォルト値をそのまま使用するか、新しい値を指定する、もしくは **Indefinite** を選択します。
11. [**Finished**] をクリックします。

Kerberos Delegation 構成オブジェクトと Kerberos Delegation プロファイルを作成した後、バーチャルサーバの [**Authentication Profile**] 設定を指定して、このプロファイルをバーチャルサーバに割り当てる必要があります。

バーチャルサーバ設定を設定する方法については、第 2 章「*バーチャルサーバの設定*」を参照してください。

11

その他のプロファイルの使用

- その他のタイプのプロファイルの概要
- OneConnect プロファイルの設定
- NTLM プロファイルの設定
- Statistics プロファイルの設定
- Stream プロファイルの設定

その他のタイプのプロファイルの概要

前の章で説明したプロファイル以外に、以下のプロファイルを設定できます。

- OneConnect
- NTLM
- 統計
- Stream

BIG-IP® では、プロファイルのタイプごとに、デフォルト設定を持つ事前設定されたプロファイルを使用できます。ほとんどの場合、これらのデフォルトプロファイルはそのまま使用できます。これらの設定を変更する場合は、プロファイルの作成時にプロファイルを設定するか、プロファイルの作成後にプロファイルの設定を変更できます。

この章の後半では、OneConnect™、NTLM、Statistics、Stream プロファイルに含まれるトラフィック管理設定を取り上げています。その他のタイプのプロファイルの設定については、次を参照してください。

- HTTP、FTP、RTSP、SIP、iSession プロファイルについては、第 6 章「*アプリケーション層トラフィックの管理*」を参照してください。
- セッションパーシステンスのプロファイルについては、第 7 章「*セッションパーシステンスの有効化*」を参照してください。
- Fast L4、Fast HTTP、HTTP Class、TCP、UDP、SCTP プロファイルについては、第 8 章「*プロトコルプロファイルの管理*」を参照してください。
- SSL プロファイルについては、第 9 章「*SSL トラフィックの管理*」を参照してください。
- アプリケーショントラフィックのリモート認証用プロファイルについては、第 10 章「*アプリケーショントラフィックの認証*」を参照してください。

OneConnect プロファイルの設定

OneConnect プロファイルは、BIG-IP でコネクションプーリングを有効にするための設定ツールです。***コネクションプーリング***は、BIG-IP が接続を処理する方法を最適化します。BIG-IP でコネクションプーリングが有効になっていると、クライアントリクエストで既存のサーバ側接続を利用できるので、このような要求を処理するためにサーバが開くサーバ側接続の数が減少することになります。

BIG-IP では、同じ OneConnect プロファイルおよび同じプールを参照する複数のバーチャルサーバからコネクションをプールできます。

◆ 重要

*コネクションプーリングを有効にするには、「OneConnect Transformations」という関連機能もあわせて有効にする必要があります。この機能は、HTTP プロファイル内から有効にします。**OneConnect 変換** HTTP プロファイル設定は、HTTP/1.0 接続に適用され、有効になっている場合、BIG-IP は、HTTP リクエスト中の **Connection** ヘッダの値を **Keep-Alive** に変換し、接続を開いたままにします。*

第 2 章「*バーチャルサーバの設定*」ですでに述べたように、BIG-IP の標準アドレス変換メカニズムは要求内の宛先 IP アドレスのみを変換し、送信元 IP アドレス（クライアントノードの IP アドレス）の変換は行いません。しかし、OneConnect 機能を有効にして、複数のクライアントノードがサーバ側のコネクションを再利用できるようにした場合、各クライアントノードのリクエストヘッダに含まれる発信元 IP アドレスは常に、サーバ側のコネクションを開始した特定のクライアントノードの IP アドレスとなります。これによってトラフィックフローが影響を受けることはありませんが、特定のシステムアウトプットを見ればそうなっていることが分かります。

表 11.1 に、OneConnect プロファイルタイプの設定を列挙して説明します。

設定	説明	デフォルト値
Name	この設定は、プロファイルの一意の名前を指定します。	デフォルト値なし
Parent Profile	この設定は、親プロファイルとして使用するプロファイルを指定します。新しいプロファイルは、指定された親プロファイルから非カスタム設定および値をすべて継承します。	**oneconnect**
Source Mask	BIG-IP では、この設定の値を送信元アドレスに適用して、再利用のための適格性を判断します。マスクを **0** に設定すると、BIG-IP では、すべてのクライアント間で再利用可能なコネクションが共有されるようになります。ホストマスク（つまり、すべての値がバイナリの **1**）を設定すると、BIG-IP では、同じクライアント IP アドレスから生成された再利用可能なコネクションのみが共有されるようになります。	**0.0.0.0**

***表 11.1** OneConnect プロファイルの設定*

設定	説明	デフォルト値
Maximum Size	この設定は、BIG-IP がコネクション再利用プールで保持するコネクションの最大数を定義します。プールがすでにいっぱいになっている場合、応答の完了後にサーバ側コネクションは終了します。 ***重要** : OneConnect プロファイルを使用する場合、**Maximum Size** 設定値には、プールメンバに設定されたコネクション数制限より少なくとも1つ以上大きな数値を指定する必要があります。これは、プールメンバのコネクションリミットの一部として、BIG-IP にコネクション再利用プールの接続が含まれるからです。したがって、プールメンバのコネクション数が許可されたコネクション数に達すると、バーチャルサーバが利用できなくなります。*	**10000**
Maximum Age	この設定は、コネクション再利用プール内で接続に許可されている最大秒数を定義します。経過時間がこの値を超えるコネクションについては、BIG-IP によって再利用プールからそのコネクションが削除されます。	**86400**
Maximum Reuse	この設定は、サーバ側コネクションを再利用できる最大回数を指定します。	**1000**
Idle Timeout Override	この設定は、コネクションフローが削除対象として適格になる前に、コネクションがアイドル状態である秒数を指定します。この設定を使用して、再利用のためにプールされた接続のタイムアウト値を増やすことができます。指定できる値は、[**Disabled**]、[**Indefinite**]、または [**Specify**]（ユーザ指定の数値）です。	**Disabled**

***表 11.1** OneConnect プロファイルの設定（続き）*

iRules™ **ONECONNECT** コマンドの詳細については、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** と、第 17 章「*iRule の記述*」を参照してください。

NTLM プロファイルの設定

NT Lan Manager（NTLM）は、ユーザのパスワードをネットワークで送信することなく、暗号化された要求 / 応答プロトコルを使ってユーザの認証を行う 業界標準テクノロジです。パスワードを送信する代わりに、認証を要求するシステムは、セキュアな NTLM クレデンシャルにアクセスしていることを証明する演算を実行します。NTLM クレデンシャルには、対話型のログインプロセス時に取得したドメイン名やユーザ名などのデータが使用されます。

BIG-IP LTM 内の NTLM プロファイルは、NT LAN Manager トラフィック処理時のネットワークパフォーマンスを最適化します。NTLM プロファイルと OneConnect プロファイルの両方がバーチャルサーバに関連付けられている場合、ローカルトラフィック管理システムはサーバ側のコネクションプーリングを利用して、NTLM 接続を管理することができます。

表 11.2 では、NTLM プロファイルタイプの設定を一覧にまとめて解説しています。

設定	説明	デフォルト値
Name	この設定は、プロファイルの一意の名前を指定します。	デフォルト値なし
Parent Profile	この設定は、親プロファイルとして使用するプロファイルを指定します。新しいプロファイルは、指定された親プロファイルから非カスタム設定および値をすべて継承します。	**ntlm**
Insert Cookie Name	システムが Cookie に挿入する Cookie 名を指定します。	**NTLMconnpool**
Insert Cookie Passphrase	システムが Cookie に挿入する Cookie パスフレーズを指定します。デフォルト値は空白であり、これはパスフレーズが挿入されていないことを示します。	デフォルト値なし
Key By Existing Cookie	既存の Cookie をキーとして使用するかどうかを指定します。	無効（チェックされていない）
Cookie Name	Cookie の名前を指定します。	**mycookie**
Key By NTLM Domain	NTLM ドメインをキーとして使用するかどうかを指定します。	無効（チェックされていない）
Key By Client IP Address	クライアント IP アドレスをキーとして使用するかどうかを指定します。	無効（チェックされていない）
Key By NTLM Target	NTLM ターゲットをキーとして使用するかどうかを指定します。	無効（チェックされていない）
Key by NTLM User	NTLM ユーザをキーとして使用するかどうかを指定します。	有効（チェックあり）
Key By NTLM Workstation	NTLM ワークステーションをキーとして使用するかどうかを指定します。	無効（チェックされていない）

***表 11.2** NTLM プロファイルの設定*

◆注

*NTLM プロファイルに加えて、OneConnect プロファイルもあわせて作成する必要があります。作成後、両方のプロファイルをバーチャルサーバに関連付けます。OneConnect プロファイルの作成手順については、「**OneConnect プロファイルの設定**」（11-2 ページ）を参照してください。*

Statistics プロファイルの設定

Statistics プロファイルは、ユーザ定義の統計情報カウンタを提供します。各プロファイルは 32 個の項目（**Field1** ～ **Field32**）を含んでおり、名前付きカウンタを定義します。Tcl ベースの iRule コマンドを使用すると、名前を使用して、トラフィックの処理中にカウンタを操作できます。

たとえば、**my_stats** という名前のプロファイルを設定できます。このプロファイルは、カウンタ **tot_users**、**cur_users**、および **max_users** をプロファイル設定 **Field1**、**Field2**、および **Field3** にそれぞれ割り当てます。この後、**track_users** という名前の iRule を作成し、**my_stats** プロファイルと **track_users** iRule をバーチャルサーバ **stats-1** に割り当てることができます。図 11.1 は、この設定を示します。

```
profile stats my_stats {
   defaults from stats
   field1 tot_users
   field2 cur_users
   field3 max_users
}

rule track_users {
   when CLIENT_ACCEPTED {
       STATS::incr my_stats tot_users
       STATS::setmax my_stats max_users [STATS::incr my_stats
cur_users]
   }
}

virtual stats-1 {
   destination 10.10.55.66:http
   ip protocol tcp
   profile http my_stats tcp
   pool pool1
   rule track_users
}
```

***図 11.1** iRule で使用される Statistics プロファイルカウンタの例*

この例では、カウンタ **tot_users** は合計コネクション数をカウントし、カウンタ **cur_users** は現在のコネクション数をカウントし、カウンタ **max_users** はカウンタ **cur_users** の最大値を保持しています。

iRules**STATS** コマンドの詳細については、F5 Networks DevCentral の Webサイト**http://devcentral.f5.com/Default.aspx?tabid=119**と、第 17 章「*iRule の記述*」を参照してください。

Stream プロファイルの設定

Stream プロファイルを使用すると、データストリーム内（TCP コネクションなど）の文字列を検索および置換できます。表 11.3 に、Stream プロファイルタイプの設定を列挙して説明します。

設定	説明	デフォルト値
Name	この設定は、プロファイルの一意の名前を指定します。	デフォルト値なし
Parent Profile	この設定は、親プロファイルとして使用するプロファイルを指定します。新しいプロファイルは、指定された親プロファイルから非カスタム設定および値をすべて継承します。	**stream**
Source	検索対象のソース文字列を指定します。	デフォルト値なし
Target	置換対象のターゲット文字列を指定します。	デフォルト値なし

***表 11.3** Stream プロファイルの設定*

リストタイプでは、パターン文字列の大文字と小文字が区別されます。たとえば、**www.f5.com** というパターン文字列は **www.F5.com** とは別のものとして扱われます。大文字と小文字の区別を無効にするには、Linux **regexp** コマンドを実行します。

iRules**STREAM** コマンドの詳細については、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** と、第 17 章「*iRule の記述*」を参照してください。

12

モニタの設定

- モニタの概要
- カスタムモニタの作成
- 特殊な設定に関する留意事項
- モニタのプールとノードへの関連付け
- モニタの管理

モニタの概要

BIG-IP® ローカルトラフィックマネージャでは、プールメンバもしくはノードの健全性やパフォーマンスを監視できるようになっています。ローカルトラフィックマネージャでは以下のモニタリング方法をサポートしています。

- *シンプルモニタリング*
  *シンプルモニタリング*は、単にノードの状態が **up** であるか **down** であるかを判断するためのものです。シンプルモニタではプールメンバを監視することはなく（つまり、ノード上の個々のプロトコル、サービス、アプリケーションは監視の対象外）、ノード自体の監視のみが行われます。同システムには、ICMP と TCP_ECHO という 2 種類のシンプルモニタが含まれます。

- *アクティブモニタリング*
  *アクティブモニタリング*は、プールメンバやノードの状態を一定間隔で継続的にチェックします。チェック中のプールメンバまたはノードが指定のタイムアウト時間内に応答しない場合や、プールメンバまたはノードの状態がパフォーマンスの低下を示している場合に、BIG-IP は別のプールメンバまたはノードにトラフィックをリダイレクトします。アクティブモニタにはさまざまな種類があり、それぞれが特定のプロトコル、サービス、アプリケーションの状態をチェックします。たとえば、モニタタイプの 1 つが HTTP です。HTTP タイプのモニタでは、プール、プールメンバまたはノードの HTTP サービスのアベイラビリティを監視できます。WMI のモニタタイプは、Windows Management Instrumentation（WMI）ソフトウェアを実行しているプール、プールメンバ、またはノードのパフォーマンスを監視します。アクティブモニタは、ECV（Extended Content Verification）モニタと EAV（Extended Application Verification）モニタのいずれかに分類されます。

- *パッシブモニタリング*
  *パッシブモニタリング*はクライアントリクエストの一環として行われます。この種のモニタリングは、一定時間内に試みられる接続やデータ要求の回数を指定して、プールメンバの健全性をチェックします。決められたインターバル内に指定された回数だけ接続や要求が試みられたにもかかわらず、サーバに接続できなかったか応答が返されなかった、もしくは誤応答が返された場合は、そのプールメンバは **down** とマーク付けされます。パッシブモニタは、***Inband*** モニタの 1 種類だけとなっています。

表 12.1（12-2 ページ）に各モニタリング方法のメリットと制約をまとめていますので、モニタリング方法を選択する際の参考にしてください。

モニタリング方法	メリット	制約
シンプル	• ノードの状態が **up** か **down** かが分かるだけでいいという場合に適している。	• ノードの健全性のみをチェックし、プールメンバのチェックは行わない。
Active	• 具体的な応答をチェックできる。 • クライアントトラフィックがあってもなくても実行可能。	• クライアントリクエストとサーバ応答を超えた余分なネットワークトラフィックが作成される。 • プールメンバを **down** とマーク付けするのに時間がかかる場合がある。
パッシブ	• クライアントリクエストとサーバ応答を超えた余分なネットワークトラフィックが作成されることはない。 • ネットワークトラフィックが存在するかぎり、プールメンバをすばやく **down** とマーク付けすることができる。	• 具体的な応答をチェックできない。 • プールメンバを **up** とマーク付けするのに時間がかかる場合がある。

***表 12.1** モニタリング方法の比較*

モニタタイプの要約

ローカルトラフィック管理システムにはさまざまなモニタが含まれており、任意のプール、プールメンバ、ノードに関連付けるモニタのタイプを選択できるようになっています。

表 12.2 では、BIG-IP で利用可能なモニタタイプを一覧にし、解説を行っています。

◆ 重要

*各モニタタイプの詳細については、付録 A「**ヘルスモニタとパフォーマンスモニタ**」を参照してください。*

.

モニタタイプ	説明
ICMP	Internet Control Message Protocol（ICMP）を使用してノードの状態をチェックします。
TCP Echo	Transmission Control Protocol（TCP）を使用してノードの状態をチェックします。
Gateway ICMP	ハイアベイラビリティのために、プール内で gateway failsafe を実装するノードをチェックします。
HTTP	Web ページからの特定コンテンツの受信を試みることで、Hypertext Transfer Protocol（HTTP）サービスを検証します。
HTTPS	Secure Socket Layer（SSL）セキュリティで保護された Web ページからの特定コンテンツの受信を試みることで、Hypertext Transfer Protocol Secure（HTTPS）サービスを検証します。
TCP	ノードからの特定コンテンツの受信を試みることで、Transmission Control Protocol（TCP）サービスを検証します。
TCP Half Open	関連付けられたサービスに TCP SYN パケットを送信して、そのサービスの監視を行います。TCP Half Open モニタは、SYN-ACK パケットを受信するとすぐに、そのサービスを **up** とマーク付けします。
External	ユーザが独自のプログラムを使用してサービスを監視できるようにします。

***表 12.2** BIG-IP で利用できるモニタタイプ*

モニタタイプ	説明
FirePass	FirePass システムの監視を行います。
FTP	BIG-IP の **/var/tmp** ディレクトリに特定のファイルをダウンロードすることで、File Transfer Protocol（FTP）サービスを検証します。ファイルは正常にダウンロードされると、保存されません。
IMAP	サーバ上の特定のメールフォルダを開くことで、Internet Message Access Protocol（IMAP）を検証します。このモニタは **pop3** モニタとほぼ同じです。
LDAP	特定のユーザの認証を試みることで、Lightweight Directory Access Protocol（LDAP）サービスを検証します。
Module Score	BIG-IP GTM と BIG-IP LTM を有効にして、Web Accelerator モジュールと Application Security Manager モジュールに関連付けられた BIG-IP LTM バーチャルサーバに均等に負荷を分散します。
MSSQL	Microsoft® Windows® SQL ベースのサービスを検証します。MSSQL タイプのモニタを使用するには、一連の Microsoft® JDBC Java™ Archive（JAR）ファイルを前もってインストールしておく必要があります。これらのファイルのインストールについては、付録第 B 章「*モニタについての追加留意事項*」を参照してください。
NNTP	サーバからニュースグループの識別文字列の検索を試みることで、Usenet News protocol（NNTP）サービスを検証します。
Oracle	サービスへの Oracle ログインの実行を試みることで、Oracle® のサービスを検証します。
POP3	指定されたユーザとしてプール、プールメンバ、またはノードへのログオンおよびログオフを試みることで、Post Office Protocol（pop3）サービスを検証します。
RADIUS	指定されたユーザの認証を試みることで、Remote Access Dial-in User Service（RADIUS）サービスを検証します。
Real Server	RealServer データコレクションエージェントを実行しているプール、プールメンバ、またはノードのパフォーマンスをチェックし、それに応じてトラフィックをダイナミックにロードバランスします。
RPC	**rpcinfo** コマンドを使って、RPC（Remote Procedure Call）サーバのアベイラビリティを検証します。
SASP	SASP（Server/Application State Protocol）を使って、IBM® Group Workload Manager により管理されているリソースのアベイラビリティの伝達・検証を行います。
SIP	デバイスの Session Initiation Protocol（SIP）Call-ID サービスの状態をチェックします。SIP プロトコルは、リアルタイムのメッセージ、音声、データ、および映像を有効にします。
SMB	Server Message Block（SMB）を採用することで、SMB/CIFS サーバもしくはそのサーバ上の特定のシェアのアベイラビリティを検証します。
SMTP	Simple Mail Transport Protocol（SMTP）の標準コマンドを発行することで、プール、プールメンバ、またはノードの状態をチェックします。
SNMP DCA	SNMP データコレクションエージェントを実行しているプール、プールメンバ、またはノードの現在の CPU、メモリ、およびディスク使用状況をチェックし、それに応じてトラフィックをダイナミックにロードバランスします。
SNMP DCA Base	SNMP データコレクションエージェントを実行しているプール、プールメンバ、またはノードの現在のユーザの使用状況をチェックし、それに応じてトラフィックをダイナミックにロードバランシングします。モニタを設定する方法によって、BIG-IP が収集するデータが決まります。
SOAP	Simple Object Access Protocol（SOAP）に基づいて Web サービスをテストします。

***表 12.2** BIG-IP で利用できるモニタタイプ（続き）*

モニタタイプ	説明
UDP	プール、プールメンバ、またはノードに UDP パケットを送信し、その返信を受け取ることを試みることによって、User Datagram Protocol（UDP）サービスを検証します。
WMI	Windows Management Infrastructure（WMI）データコレクションエージェントを実行しているプール、プールメンバ、またはノードのパフォーマンスをチェックし、それに応じてトラフィックをダイナミックにロードバランスします。
Inband	一定時間内に接続またはデータ要求に何度失敗できるかを指定し、 その回数に基づきプールメンバのアベイラビリティを特定します。

***表 12.2** BIG-IP で利用できるモニタタイプ（続き）*

モニタの設定について

どのモニタにも値が設定されています。この設定と値は、モニタタイプにより異なります。場合によっては、BIG-IP がデフォルト値を割り当てます。たとえば図 12.1 は、これらの設定とデフォルト値を持つ ICMP タイプのモニタを示します。

```
Name my_icmp
Type ICMP
Interval 5
Timeout 16
Transparent No
Alias Address * All Addresses
```

***図 12.1** デフォルト値が設定されたモニタの例*

図 12.1 の設定では、ICMP タイプのモニタが 5 秒ごとに IP アドレスの状態をチェックし、それぞれ 16 秒後にタイムアウトするよう指定しています。モニタがチェックする、宛先 IP アドレスは **Alias Address** 設定で指定され、その値は *** All Addresses** です。このように、前述した例ではモニタが関連付けられているすべての IP アドレスがチェックされます。モニタ設定の詳細については、「*特殊な設定に関する留意事項*」（12-10 ページ）を参照してください。

モニタ実装の概要

モニタの実装は、設定ユーティリティまたはコマンドラインユーティリティを使って行います。設定済みモニタを使用するかカスタムモニタを使用するかによって、モニタ実装のタスクは異なります。***設定済みモニタ***は、BIG-IP が提供する設定済みの既存のモニタです。***カスタムモニタ***とは、使用可能なモニタタイプの 1 つをベースに作成するモニタのことです。

設定済みのモニタを実装する場合は、モニタをプール、プールメンバ、またはノードに関連付けて、関連するプールを参照するようにバーチャルサーバを設定するだけでよくなっています。カスタムモニタを実装する場合は、最初にそのカスタムモニタを作成する必要があります。次に、そのカスタムモニタをプール、プールメンバ、またはノードと関連付けて、そのプールを参照するようにバーチャルサーバを設定します。

カスタムモニタの作成手順については「*カスタムモニタの作成*」（12-8 ページ）を、モニタをプールやプールメンバ、またはノードと関連付ける手順については「*モニタのプールとノードへの関連付け*」（12-15 ページ）を、それぞれ参照してください。

プールを参照するようにバーチャルサーバを設定する方法については、第 2 章「*バーチャルサーバの設定*」で紹介しています。

設定済みモニタの使用

BIG-IP では、一部のモニタタイプに一連の設定済みモニタが用意されています。設定済みモニタはそのままの状態で使用されるように設計されているため、設定を変更することはできません。設定済みモニタは、明示的にモニタを作成する手間を省くために用意されています。設定の値がニーズに合っている場合は設定済みモニタを利用してください。

BIG-IP に用意されている設定済みモニタの名称は、以下のとおり です。

- **gateway_icmp**
- **http**
- **https**
- **https_443**
- **icmp**
- **inband**
- **real_server**
- **snmp_dca**
- **tcp**
- **tcp_echo**

設定済みモニタの例として **icmp** モニタについて説明します。図 12.2 は、**icmp** モニタ、および設定されている **Interval**、**Timeout**、および **Alias Address** 設定の値を示します。**Interval** 値は **5**、**Timeout** 値は **16**、Transparent 値は **No**、そして **Alias Address** 値は ***All Addresses** です。

```
Name icmp
Type ICMP
Interval 5
Timeout 16
Transparent No
Alias Address * All Addresses
```

図 12.2 icmp 設定済みモニタ

Interval、**Timeout**、**Transparent**、および **Alias Address** 値がニーズに合う場合は、設定ユーティリティ内の [**Pools**] または [**Nodes**] 画面を使用して、**icmp** 設定済みモニタをプール、プールメンバ、またはノードに直接割り当ててください。この場合、単に設定済みのモニタ設定の値を表示する場合を除いて、[Monitors] 画面を使用する必要はありません。

◆ 重要

設定済みモニタは、すべて ***Common*** *パーティションに置かれます。パーティションの詳細については、『****TMOS™ Management Guide for BIG-IP® Systems****』を参照してください。*

設定済みモニタに設定されている値を使用しない場合は、カスタムモニタを作成します。

カスタムモニタの使用

設定済みモニタで定義された値がニーズに合わない場合、もしくは設定済みモニタの中に作成しようとしているタイプのものが含まれない場合に、カスタムモニタを作成します。モニタタイプの詳細については、「*モニタタイプの要約*」（12-2 ページ）を参照してください。

設定済みモニタからの設定のインポート

作成するタイプのカスタムモニタに対応した設定済みモニタが存在する場合、その設定済みモニタの設定と値をカスタムモニタにインポートできます。その後、これらの設定値をニーズに合わせて自由に変更できます。たとえば、**my_icmp** と呼ばれるカスタムモニタを作成すると、モニタは設定済みモニタ **icmp** の設定と値を継承できます。既存の設定値をインポートできるこの機能は、新しいモニタに合わせて一部の設定値を保持し、残りの設定値を変更する場合に便利です。

図 12.3（12-6 ページ）は、設定済みモニタ **icmp** に基づく **my_icmp** と呼ばれる ICMP タイプのカスタムモニタの例を示します。

Interval 値は **10** に、**Timeout** 値は **20** にそれぞれ変更されています。その他の設定は設定済みモニタで定義されている値をそのまま使用しています。

```
Name my_icmp
Type ICMP
Interval 10
Timeout 20
Transparent No
Alias Address * All Addresses
```

図 12.3 *設定済みモニタに基づいたカスタムモニタ*

カスタムモニタからの設定のインポート

設定済みモニタの代わりに別のカスタムモニタから設定をインポートできます。これは、別のカスタムモニタに定義されている設定値を使用する場合や、作成するモニタタイプの設定済みモニタが存在しない場合に便利です。たとえば、**my_oracle_server2** と呼ばれるカスタムモニタを作成する場合、**my_oracle_server1** などの既存の Oracle タイプのモニタから設定をインポートできます。この場合、BIG-IP には設定済みの Oracle タイプのモニタがないため、カスタムモニタは設定値をインポートできる唯一のモニタの種類です。

モニタの選択はごく簡単です。**icmp** と同様に、モニタはそれぞれモニタがチェックするサービスの種類に基づいた **Type** 設定（**http**、**https**、**ftp**、**pop3** など）を持ち、その種類が名前になっています。(ユーザ指定のプログラムを呼び出す **external** モニタのようなポート固有のモニタは例外です)。

モニタの選択と設定の詳細については、「*カスタムモニタの作成*」(12-8 ページ）を参照してください。

カスタムモニタの作成

カスタムモニタを作成する場合は、設定ユーティリティまたはコマンドラインユーティリティを使用してモニタに固有の名前を指定し、モニタのタイプを指定します。そのタイプがすでに存在する場合は、設定とその値を既存のモニタからインポートします。その後、インポートされた設定の値を変更できます。

各カスタムモニタはモニタのタイプを基に作成する必要があります。モニタを作成すると、設定ユーティリティによってモニタタイプのリストが表示されます。モニタタイプを指定するには、チェックするサービスに対応したタイプを選択します。たとえば、プールでのHTTPサービスの健全性をチェックするモニタを作成する場合はモニタタイプとして **HTTP** を選択します。

プールまたはプールメンバで複数のサービス（たとえば **HTTP** と **HTTPS**）をチェックする場合、そのプールまたはプールメンバに複数のモニタを関連付けることができます。詳細については、第 4 章「*ロードバランシングプールの設定*」を参照してください。

モニタを実装する理由はサービスをチェックするためだけではありません。宛先 IP アドレスがアクティブであること、または透過ノードを介した宛先 IP アドレスへのパスがアクティブであることを検証する場合は、単純なモニタの **icmp** または **tcp_echo** のいずれかを使用します。または、TCP のみを検証する場合は、モニタ **tcp** を使用します。

◆注

*カスタムモニタを作成する前に、モニタのタイプを決定する必要があります。モニタタイプの詳細については、付録 A「**ヘルスモニタとパフォーマンスモニタ**」を参照してください。*

◆重要

*カスタムモニタが作成されると、BIG-IP はそのモニタを現在の管理パーティションに配置します。管理パーティションの詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

カスタムモニタを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Monitors**] をクリックします。
   [Monitors] 画面が開きます。
2. 画面の右上で [**Create**] をクリックします。
   [New Monitor] 画面が開きます。

   ***注**: [Create] ボタンをクリックできない場合は、現在のユーザロールではカスタムモニタを作成する権限が与えられていないことを示します。*
3. [**Type**] 設定から作成するモニタタイプを選択します。
   そのタイプが既に存在する場合は、[**Import Settings**] が表示されます。

4. 以下のいずれかの設定に基づいて名前を指定します。
   - [**Import Settings**] が表示された場合は、このリストからモニタ名を選択します。
   - 選択したモニタのタイプが [**Name**] ボックスに存在しない場合は、カスタムモニタの一意の名前を入力します。
5. 画面の [Configuration] セクションの [**Advanced**] を選択します。これによって、追加のデフォルト値を変更できます。
6. 表示されているすべての設定を行います。
7. [**Finished**] をクリックします。

◆ 重要

カスタムモニタの値を定義する場合は、予約済みキーワードの一覧にある値の使用は避けてください。

特殊な設定に関する留意事項

設定済みモニタまたはカスタムモニタにはすべて、デフォルト値が割り当てられた設定があります。これらのデフォルト値を変更する際には、以下のトピックが役に立ちます。

- **宛先の設定**
  詳細については、次の項の「*宛先の設定*」を参照してください。
- **Transparent モードと Reverse モード**
  詳細については、「*Transparent モードと Reverse モード*」(12-11 ページ）を参照してください。
- **Manual Resume の設定**
  詳細については、「*Manual Resume の設定*」（12-12 ページ）を参照してください。
- **Check Until Up の設定**
  詳細については、「*Check Until Up の設定*」（12-14 ページ）を参照してください。

宛先の設定

デフォルトでは、モニタの **Alias Address** 設定の値は、ワイルドカード *** Addresses** に設定され、**Alias Service Port** 設定はワイルドカード ***Ports** に設定されています。この値によって、プール、プールメンバ、またはノードに作成されるモニタインスタンスは、そのノードのアドレス、またはアドレスとポートを宛先としてとります。ただし、カスタムモニタを作成することで、どちらかまたは両方のワイルドカード記号を明示的な宛先値と置き換えることができます。**Alias Address** または **Alias Service Port**、あるいはその両方の設定の明示的な値は、プール、プールメンバ、またはノードの値とは異なる特定のアドレスまたはポート、あるいはその両方にインスタンスの宛先を強制するために使用されます。

ECV モニタ **http**、**https**、および **tcp** には、**Send String**（文字列の送信）および **Receive String**（文字列の受信）があります。

最も一般的な **Send String** 値は、**GET /** であり、これは Web サイトのデフォルトの HTML ページを検索します。Web サイトから特定のページを検索するには、次の完全修飾パス名である **Send String** 値を入力できます。

```
"GET /www/support/customer_info_form.html"
```

Receive String 式は、返されたリソースからモニタが検索するテキスト文字列です。最も一般的な **Receive String** 式は、サイト上の特定の HTML ページに組み込まれているテキスト文字列を含みます。テキスト文字列には通常のテキスト、HTML タグ、または画像名を指定できます。

以下のサンプルの **Receive** 式では、標準の HTML タグが検索されます。

```
"<HEAD>"
```

また、デフォルトの Null の **Receive String** 値 **[""]** も使用できます。この場合、いずれのコンテンツ検索も一致すると見なされます。**Send String** と **Receive String** の両方を空のままにすると、単純な接続の確認のみが実行されます。

HTTP モニタと FTP モニタについては、**Send String** ステートメントと**Receive String**ステートメントの代わりに特殊な設定**get**または**hurl**を使用できます。FTP モニタに限っては、**GET** 設定によって検索するファイルへのフルパスを指定します。

Transparent モードと Reverse モード

モニタの通常およびデフォルトの動作は、指定されていないルートによって宛先プール、プールメンバ、またはノードに対して ping コマンドを発行すること、およびテストが正常終了した場合はノードを **up** としてマークすることです。ただし、特定のモニタタイプでは、モニタが宛先サーバに ping コマンドを発行するルートを指定できます。これは、カスタムモニタ内に **Transparent** または **Reverse** 設定を指定することで設定できます。

◆ 重要

IPv6 アドレッシングもしくはルートドメイン機能を使用している場合は、Transparent 機能を含むモニタを使用することはできません。

- **Transparent 設定**
  透過プール、プールメンバ、またはノードを介して別名の宛先に ping を発行する必要がある場合もあります。カスタムモニタを作成し、**Transparent** 設定を **[Yes]** に設定すると、BIG-IP はプール、プールメンバ、またはノードに関連付けられている（通常はファイアウォール）プール、プールメンバ、またはノードを*介して* ping を発行するようにモニタに強制します。（つまり、ロードバランシングプールに 2 つのファイアウォールがある場合、宛先のプール、プールメンバ、またはノードには常に指定されたプール、プールメンバ、またはノードを介して ping コマンドが発行されます。ロードバランシング方式によって選択されたプール、プールメンバ、またはノードから発行されることはありません。）このように透過プール、プールメンバ、またはノードはテストされ、応答がないと、透過プール、プールメンバ、またはノードは **down** としてマークされます。

  ファイアウォールを介したルータのチェック、またはメールや FTP サーバのチェックなどが一般的な例です。たとえば、透過ファイアウォール **10.10.10.101:80** を介してルーターアドレス **10.10.10.53:80** をチェックするとします。これを実行するには、モニタの宛先アドレスとして **10.10.10.53:80** を指定している **http_trans** と呼ばれるモニタを作成し、**[Transparent]** 設定を **[Yes]** に設定します。次にモニタ **http_trans** を透過プール、プールメンバ、またはノードに関連付けます。

  これによって、モニタは **10.10.10.101:80** を通じて**アドレス 10.10.10.53:80** をチェックします。（言い換えれば、BIG-IP は **10.10.10.53:80** のチェックを **10.10.10.101:80** へとルーティング

します）正しい応答が **10.10.10.53:80** から受信されない場合、**10.10.10.101:80** は **down** しているとしてマークされます。プールメンバまたはノードにモニタを関連付ける方法については、「*モニタのプールとノードへの関連付け*」（12-15 ページ）を参照してください。

- **Reverse 設定**
  [**Reverse**] 設定を [**Yes**] に設定すると、モニタはテストの正常終了時にプール、プールメンバ、またはノードを **down** としてマークします。たとえば、Web サイトのホームページ上のコンテンツがダイナミックで頻繁に変更される場合は、文字列 **"Error"** を検索するリバース ECV サービスチェックを設定できます。この文字列の一致は、Web サーバが **down** であったことを意味します。

図 12.3 は、**Transparent** 設定、**Reverse** 設定、またはその両方を含むモニタを示します。

モニタタイプ	設定	
TCP	Transparent	Reverse
HTTP	Transparent	Reverse
HTTP		Reverse
TCP Echo	Transparent	
ICMP	Transparent	

表 12.3 Transparent 設定または Reverse 設定を含むモニタ

Manual Resume の設定

デフォルトでは、モニタがリソース（ノードやプールメンバ）の利用が不可能であることを検出すると、BIG-IP はそのリソースを **down** とマーク付けして、アクティブなロードバランシング方式の指示どおりに、トラフィックを次の適切なリソースに割り当てます。次にモニタがそのリソースが再び利用可能になったと判断すると、BIG-IP はそのリソースを **up** とマーク付けし、その後すぐに、接続要求のロードバランシングに利用できるとみなします。このプロセスはほとんどのリソースで有効に機能しますが、BIG-IP がリソースの利用が可能であることを自動で判断するのではなく、手動で指定したほうがいい場合もあります。リソースの利用が可能であることを手動で指定するには、モニタの **Manual Resume** を設定します。

たとえば、Web サイト用の **index.html** という HTML ファイルのアベイラビリティを監視するために、リソースにモニタを割り当てているとします。営業時間中に、あなたはその Web サイトをホスティングしているシステムを再起動する必要があると判断しました。モニタはシステムの再起動を検知し、そのリソースが利用できなくなったことをBIG-IPに通知します。システムが再起動すると、モニタは**index.html** ファイルの利用が可能であることを検知し、Web サイトへの接続要求の送信を開始します。しかし、Web サイトのそれ以外の部分では

接続要求を受信する準備が整っていないため、結果的に BIG-IP は、Web サイトがきちんと応答できる状態になる前に、接続要求を送信してしまうことになります。

このような問題を防止するには、モニタの **Manual Resume** を設定します。**Manual Resume** を **Yes** と設定すると、ユーザが手動でリソースを有効にするまで、BIG-IP はそのリソースの利用が不可能であるとみなします。

つまり、モニタの **Manual Resume** を **Yes** と設定して、そのモニタをリソースに関連付けた後に、そのリソースの利用が不可能になった場合、手動で有効にするまでリソースはオフラインのままとなります。

既存モニタの Manual Resume 設定の変更

カスタムモニタの作成時に **Manual Resume** を設定するか、もしくは既存のカスタムモニタの **Manual Resume** 設定を変更することができます。カスタムモニタの作成方法については、「*カスタムモニタの作成*」（ 12-8 ページ）を参照してください。既存のカスタムモニタの **Manual Resume** 設定を変更する場合は、以下の手順に従ってください。

Manual Resume プロパティを変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して、[**Monitors**] をクリックします。
   メインモニタ画面が開きます。
2. 該当するモニタの名前をクリックします。
   そのモニタのプロパティ画面が開きます。
3. [**Configuration**] リストから、[**Advanced**] を選択します。
4. [**Manual Resume**] オプションで、[**Yes**] をクリックします。
5. [**Update**] をクリックします。

接続の再開

モニタが **down** とマーク付けしたリソース（ノードやプールメンバなど）があり、そのリソースがその後再び利用可能になった場合、モニタの [**Manual Resume**] が **Yes** と設定されていれば、そのリソースを手動で再度有効にする必要があります。リソースを手動で再度有効にすることで、BIG-IP はそのリソースへの接続の送信を再開します。

リソースを手動で再度有効にする手順は、そのリソースがプール、プールメンバ、ノードのどれであるかによって変わってきます。詳細については、第 3 章「*ノードの設定*」と第 4 章「*ロードバランシングプールの設定*」を参照してください。

Check Until Up の設定

1 つのプールメンバに対してアクティブとパッシブの両方のモニタリングを実装する場合、アクティブモニタの **Check Until Up** 設定を有効にすることができます。アクティブとパッシブのモニタリングを設定して、**Check Until Up** 設定を有効にすると、アクティブモニタはプールメンバが **up** であることが確定した時点でそのプールメンバのヘルスチェックを終了します。プールメンバが **up** であることが確定すると、そのプールメンバに対するヘルスチェック機能は無効になり、Inband モニタが **down** 状態のプールメンバの検出を開始します。

表 12.4 では、パッシブモニタリングが設定されているときに、アクティブモニタで **Check Until Up** 機能を有効にした場合と無効にした場合の違いを比較しています。

Check Until Up 設定	Inband モニタがプールメンバを up とマーク付け	Inband モニタがプールメンバを down とマーク付け
	その場合	その場合
有効	Inband モニタがプールメンバーを **down** とマーク付けするまで、アクティブモニタはそのプールメンバのヘルスチェックを控える。	プールメンバーが **up** であるとの報告があるまで、アクティブモニタはそのプールメンバのヘルスチェックを行う。その後、Inband モニタを使ってプールメンバの監視が行われる。
無効	アクティブモニタはプールメンバのヘルスチェックを継続的に行う。	アクティブモニタはプールメンバのヘルスチェックを継続的に行う。

***表 12.4** パッシブモニタリング有効時に **Check Until Up** 設定がアクティブモニタに与える影響*

◆ 重要

*この設定を Inband モニタと共に使用する場合は、Inband モニタの **[Retry Time]** を **0** に設定してください。*

モニタのプールとノードへの関連付け

モニタを作成してその設定を構成したら、監視対象のサーバにモニタを関連付けることが最後の作業となります。サーバはモニタのタイプによってプール、プールメンバ、またはノードになります。

モニタとサーバは以下のいずれかの方法で関連付けることができます。

- **モニタとプールの関連付け**
  このタイプの関連付けでは、モニタがロードバランシングプール全体に関連付けられます。この場合、モニタはプールのすべてのメンバをチェックします。たとえば、プール **my_pool** の各メンバにモニタ **http** のインスタンスを作成できます。これによってそのプールのメンバすべてが確実にチェックされます。
- **モニタとプールメンバの関連付け**
  このタイプの関連付けでは、モニタが個別のプールメンバ、つまり IP アドレスとサービスに関連付けられます。この場合、モニタでは目的のプールメンバのみがチェックされ、プールのその他のメンバはチェックされません。たとえば、**my_pool** のプールメンバ **10.10.10.10:80** に対してモニタ **http** のインスタンスを作成できます。
- **モニタとノードの関連付け**
  このタイプの関連付けでは、モニタが特定のノードに関連付けられます。この場合、モニタではノード自体がチェックされるだけで、そのノード上で動作しているサービスはチェックされません。たとえば、ノード **10.10.10.10** に対してモニタ **icmp** のインスタンスを作成できます。この場合、モニタでは特定のノードのみがチェックされ、そのノード上で動作しているサービスはチェックされません。
  モニタは、BIG-IP で 1 つ以上のノードと関連付けるデフォルトモニタとして指定できます。この場合、モニタを特に割り当てていないノードは、デフォルトモニタを継承します。

モニタのタイプによっては、プールまたはプールメンバではなく、ノードのみに関連付けるように設計されているものもあれば、ノードではなくプールとプールメンバに関連付けるように設計されているものもあります。ノードのみのモニタは、サービスポートなしの IP アドレスだけの形式で宛先アドレスを指定します（たとえば、**10.10.10.2**）。反対に、ノード、プールおよびプールメンバに関連付けられるモニタは、IP アドレスとサービスポートの形式で宛先アドレスを指定します（たとえば、**10.10.10.2:80**）。したがって、設定ユーティリティを使用してモニタをプール、プールメンバ、ノードに関連付ける場合、該当するサーバとの関連付けに対応した設定済みモニタのみが表示されるようになっています。たとえば、モニタ **icmp** をプールまたはそのメンバに関連付けることはできません。これは **icmp** モニタがノード自身の状態をチェックするよう設計されており、そのノード上で実行しているサービスをチェックするようには設計されていないためです。

モニタをサーバに関連付けると、BIG-IP は自動的にそのサーバに対するモニタの***インスタンス***を作成します。次に、モニタの関連付けによってユーザが指定する各サーバに対してモニタのインスタンスが作成されます。したがって、サーバ上で同じモニタの複数のインスタンスが動作することがあります。

設定ユーティリティでは、サーバ上で動作しているモニタのインスタンスを無効にすることができます。これによって、実際にモニタの関連付けを削除せずにヘルスチェックやパフォーマンスチェックを延期できます。サーバの監視を再び開始する準備が整ったら、モニタのそのインスタンスを単純にもう一度有効にします。

プールとプールメンバにモニタを関連付ける方法については、第 4 章「*ロードバランシングプールの設定*」を参照してください。ノードにモニタを関連付ける方法については、第 3 章「*ノードの設定*」を参照してください。

モニタの管理

既存のモニタを管理する場合、これらのモニタを表示または削除したり、モニタのインスタンスを有効および無効にしたりできます。モニタを削除する前に、モニタの既存の関連付けをすべて削除する必要があります。

◆注

ユーザロールとパーティションアクセス割り当てに基づき、管理が許可されているモニタのみを操作することができます。

モニタを表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Monitors**] をクリックします。
   [Monitors] 画面が開きます。
2. モニタ名をクリックします。
   モニタ設定とその値が表示されます。

モニタを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Monitors**] をクリックします。
   [Monitors] 画面が開きます。
2. 削除するモニタの [Select] ボックスをチェックします。
3. [**Delete**] をクリックします。
   インターフェイス削除の確認メッセージが表示されます。
4. [**Delete**] をクリックします。

モニタインスタンスを有効または無効にするには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Monitors**] をクリックします。
   [Monitors] 画面が開きます。
2. リストからモニタ名をクリックします。
3. メニューバーの [**Instances**] をクリックします。
   既存のモニタのインスタンスが一覧表示されます。
4. 管理するインスタンスの [Select] ボックスをチェックします。
5. [**Enable**] または [**Disable**] をクリックします。
6. [Update] をクリックします。

◆注

*モニタのインスタンスはパーティション分割オブジェクトではないため、ユーザは関連するプールやプールメンバの管理が許可されていなくても、モニタのインスタンスを有効化 / 無効化することができます。たとえば、**Manager** ロールを持ち、**AppA** パーティションにのみ*

*アクセスできるユーザは、**Common** パーティション内にあるプールのインスタンスについても有効化/無効化することができます。ただし、そのモニタに関連付けられたプールやプールメンバに対して処理を実行することはできません。これは正しい機能なのですが、ユーザにとっては予想外の挙動かもしれません。モニタインスタンスに関連付けられたすべてのプールとプールメンバを同じパーティションに配置することで、このような予想外の挙動を防止することができます。*

13

NAT の設定

- ネットワークアドレス変換（NAT）の概要
- NAT の作成
- NAT の管理

ネットワークアドレス変換（NAT）の概要

NATを使うケースとして、外部ネットワークのクライアントが特定の内部ノードに要求を直接送信（つまり、通常のロードバランシングによるサーバ選択を迂回）できるようにした方がいいケースがあります。要求を内部サーバに直接送信するには、通常はクライアントが内部ノードのIPアドレス（ほとんどの場合、プライベートクラスのIPアドレス）を知っている必要があります。プライベートクラスのIPアドレスはルーティングが不可能なことから、代わりにネットワークアドレス変換（NAT）を作成して対応します。***NAT***は、外部ノードが内部ノードとのトラフィックの送受信に使用するためのルーティング可能なIPアドレスを提供します。

NATとは？

NATとは、パケットヘッダ内の1つのIPアドレスを別のIPアドレスに変換するようにBIG-IPに指示するアドレス変換オブジェクトです。NATでは、パブリックIPアドレスと内部のプライベートクラスIPアドレスが1対1でマッピングされます。

NATには以下の2種類の使用法があります。

- **プライベートクラスの宛先アドレスをパブリックアドレスに変換（外部→内部）**
  外部ノードがNATで定義されたパブリックIPアドレスにトラフィックを送信すると、BIG-IPはその宛先アドレスを関連するプライベートクラスIPアドレス（内部ネットワークの特定ノードを表す）に自動で変換します。この変換は、トラフィックを送信した外部ノードからは見えないようになっています。詳細については、「*受信接続用のNATについて*」（13-2ページ）を参照してください。
- **プライベートクラスの送信元アドレスをパブリックアドレスに変換（内部→外部）**
  NATでは、内部ノードのプライベートクラス送信元IPアドレスをパブリックIPアドレスに変換することもできます。この変換は、トラフィックを受信する外部ノードからは見えないようになっています。詳細については、「*発信接続用のNATについて*」（13-4ページ）を参照してください。

NATは、プライベートクラスのIPアドレスを持つノードでパケットを送受信するための、ルーティング可能なアドレスを提供します。

NATを作成すると、プライベートクラスのIPアドレスを1つにかぎり特定のパブリックIPアドレスに割り当てることができます。つまり、NATでは常に、プライベートクラスIPアドレスとパブリックIPアドレスが1対1でマッピングされるようになっています。複数のプライベートクラスIPアドレス（つまり複数の内部ノード）を1つのパブリックIPアドレスにマッピングしたい場合は、NATではなくSNATを作成します。詳細については、第14章「*SNATの設定*」を参照してください。

NAT を作成するには、設定ユーティリティか **bigpipe** ユーティリティのいずれかを使用します。設定ユーティリティの使い方については「*NAT の作成*」（13-6 ページ）を、**bigpipe** ユーティリティの使い方については『***Bigpipe Utility Reference Guide***』を、それぞれ参照してください。

◆注

NAT はポート変換をサポートしていないため、FTP、NT Domain、CORBA IIOP など、パケットに IP を埋め込むプロトコルには適していません。

◆ヒント

*NAT を使って内部ノードへのアクセスを提供する場合、その内部ノードのすべてのポートが解放されます。このようなセキュリティ上のリスクを緩和するには、SNAT の使用を検討してください。詳細については、第 14 章「**SNAT の設定**」を参照してください。*

BIG-IP では、受信接続か発信接続のいずれか一方に NAT を適用することができます。

受信接続用の NAT について

NAT における***受信***接続とは、外部ネットワークのノードによって開始され、BIG-IP が受信した後に、内部ネットワークのノードに到達する接続を指します。

NAT を使用しない場合

通常、BIG-IP が受信するトラフィックについては、プールで設定されたロードバランシング方式に従い、プール内のサーバに以下のような方法で負荷分散が行われます。

- 外部ネットワークのクライアントは通常、BIG-IP のバーチャルサーバにトラフィックを送信します。この場合、宛先 IP アドレスはバーチャルサーバのアドレスとなります。
- 通常バーチャルサーバは、パケットを受信すると、そのパケットの負荷を分散するために、宛先 IP アドレスをプールメンバの IP アドレスに変換します。
- 続いてプールメンバが、サーバノードのルーティングテーブルで指定された経路（理想は、内部 VLAN に割り当てられたフローティング IP アドレス）を使って、BIG-IP 経由で応答を返信します。その応答を受信すると、BIG-IP は逆変換を実行します。つまり、プールメンバの実際の送信元アドレスが、バーチャルサーバのアドレスに変換されることになります。その結果、クライアントに返される応答内の送信元アドレスはバーチャルサーバのアドレスとなり、クライアントにはこの送信元アドレスが示されます。

このような一般的なロードバランシングのシナリオによって、負荷分散されたトラフィックでは、内部ノードの内部プライベートクラス IP アドレスがクライアントシステムに示されることはありません。

NAT を使用した場合

ロードバランシングメカニズムを迂回して、クライアントシステムが内部ネットワークの特定ノードに直接パケットを送信するようにするには、ルーティング可能な IP アドレスを使ってそのサーバノードにパケットを送信する必要があります。

NAT は、クライアントが内部サーバに対して直接要求を行うためのルーティング可能なアドレスを提供することで、この問題を解消します。その際 NAT は、バーチャルサーバがプールメンバへの接続の負荷分散時に行うのと同様のアドレス変換を実行します。ただし、NAT の場合はクライアントが特定のノードに要求を送信することから、負荷分散が発生することはありません。NAT は、要求内のパブリック宛先 IP アドレスを内部ノードのプライベートクラス IP アドレスに変換します。

サーバノードから応答が送信されると、BIG-IP はバーチャルサーバと同じように逆変換を実行します（この逆変換の詳細については、「*NAT を使用しない場合*」（13-2 ページ）を参照してください）。

◆注

BIG-IP が NAT 接続の追跡を行うことはありません。したがって、NAT で指定するパブリック IP アドレスを、仮想アドレスや SNAT アドレスと同じアドレスにすることはできません。

NAT の作成例

内部ネットワークのノード（ロードバランシングサーバなど）が、**172.16.20.3** というプライベートクラスの IP アドレスを持っているとします。選択したパブリック宛先アドレス（**207.10.1.103** など）をプライベートクラスのアドレス **172.16.20.3** に変換するための NAT を作成することができます。結果、外部ネットワークのノードがアドレス **207.10.1.103** への接続を開始すると、BIG-IP は常にそのパブリック宛先アドレスをプライベートクラスのアドレス **172.16.20.3** に変換するようになります。

図 13.1 は、受信接続で発生するアドレス変換を解説したものです。

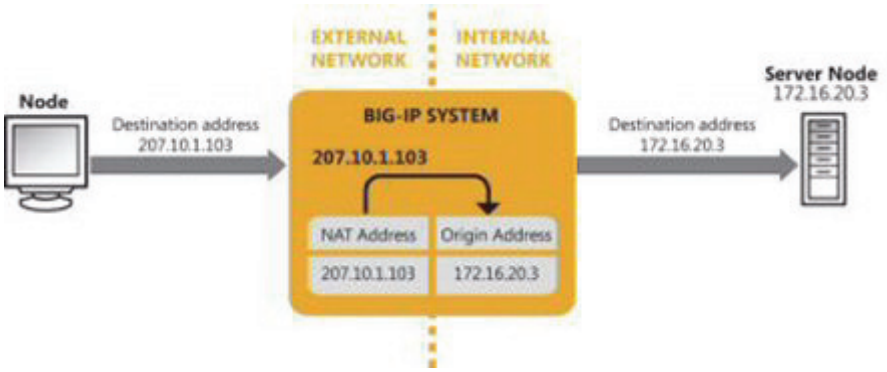

***図 13.1** 受信接続用の NAT の例*

上の例では、外部ノードが内部ノードへの接続を開始できるように、NAT はルーティング可能なアドレスを提供しています。

NAT を作成する際には、**NAT Address** と **Origin Address** の 2 つの値を指定する必要があります。この例の場合、以下のようになります。

- NAT Address は **207.10.1.103**、Origin Address は **172.16.20.3** です。
- 接続は受信接続です。つまり、外部ネットワークで開始された接続が、BIG-IP を経由して、内部ネットワークに送信されています。
- BIG-IP は、NAT Address を Origin Address に変換します。
- NAT Address と Origin Address は共に宛先アドレスです。

発信接続用の NAT について

前の「*受信接続用の NAT について*」（13-2 ページ）では、BIG-IP が通常どのように着信トラフィックの負荷を分散し、応答内の発信元 IP アドレスを仮想アドレスに変換するのかについて説明しました。

しかし、場合によっては、内部ノードは単に要求に応答するだけでなく、接続を開始しなければならないケースも出てきます。内部ネットワークのノードによって接続が開始された場合、その接続は***発信***接続と見なされます。この場合、発信パケットは負荷分散された要求への応答とはならないため、バーチャルサーバを通過することはなく、負荷分散時の送信元 IP アドレス変換が実行されることもありません。

NAT を使用しなければ、送信元 IP アドレスはルーティング不可能なアドレスとなります。しかし、NAT を使用した場合、BIG-IP は内部ノードのプライベートクラス IP アドレスをパブリック IP アドレスに変換し、外部ノードはそのアドレスに対して応答を送信することができます。

NAT の作成例

内部ノード（メールサーバなど）が **172.16.20.1** というプライベートクラスの IP アドレスを持っているとします。このプライベートクラスアドレス **172.16.20.1** を、選択したパブリック送信元スアドレス（**207.10.1.101** など）に変換するための NAT を作成することができます。結果、内部ノード **172.16.20.1** が外部ネットワークのノードへの接続を開始すると、BIG-IP は常に発信元アドレス **172.16.20.1** をパブリックアドレス（**207.10.1.101**）に変換するようになります。

図 13.2（13-5 ページ）は、発信接続で発生するアドレス変換を解説したものです。

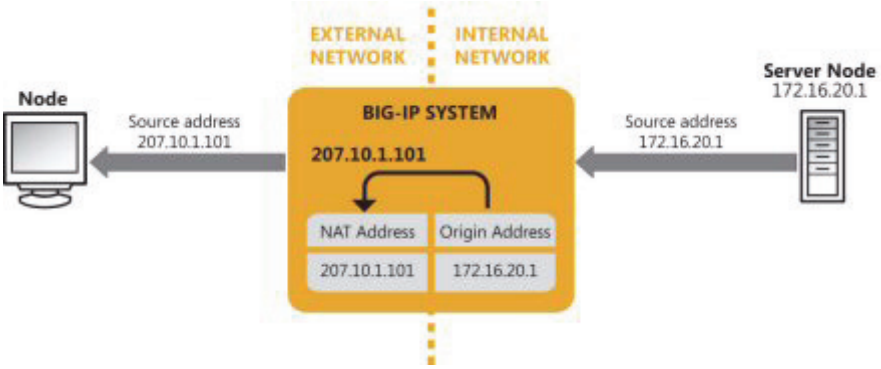

図 13.2 発信接続用の NAT の例

上の例において、NAT は、送信元アドレスとしてのプライベートクラス IP アドレスを示すことなく、内部ノードが外部ネットワークのノードへの接続を開始する方法を提供しています。

すでに述べたように、NAT では **NAT Address** と **Origin Address** の 2 つの値を指定する必要があります。この例の場合、以下のようになります。

- NAT Address は **207.10.1.101**、Origin Address は **172.16.20.1** です。
- 接続は発信接続です。つまり、内部ネットワークで開始された接続が、BIG-IP を経由して、外部ネットワークに送信されています。
- BIG-IP は、Origin Address を NAT Address に変換します。
- Origin Address と NAT Address は共に送信元アドレスです。

すでに説明したように、NAT では常に、パブリックアドレスとプライベートクラスアドレスが 1 対 1 でマッピングされるようになっています。複数の内部ノードを 1 つのパブリックアドレスにマッピングしたい場合は、NAT ではなくセキュアネットワークアドレス変換（SNAT）を使用します。SNAT は発信接続に対してのみ利用可能です。詳細については、第 14 章「*SNAT の設定*」を参照してください。

NAT の作成

設定ユーティリティを利用して、内部ノードごとに別々の NAT アドレスを作成する必要があります。NAT を作成するときに、一連のプロパティを設定します。NAT を作成する際に [**NAT Address**] 設定と [**Origin Address**] 設定を行う必要がありますが、その他の設定にはデフォルト値を使用したり、後でその値を変更したりできます。

NAT を作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開し、[**SNATs**] をクリックします。
   [SNATs] 画面が開きます。
2. メニューバーで、[**NAT List**] をクリックします。
3. 右上の [**Create**] をクリックします。
   [New NAT] 画面が開きます。

   ***注 : [Create]** ボタンをクリックできない場合は、現在のユーザロールに NAT を作成する権限がないということになります。*
4. [**NAT Address**] ボックスに、内部ノードのルーティング可能なパブリックアドレスとして使用する IPアドレスを入力します。
5. [**Origin Address**] ボックスに、その内部ノードのプライベートクラス IP アドレスを入力します。
6. その他すべての値を、そのまま使用するか、必要に応じて変更します。
7. [**Finished**] をクリックします。

表 13.1 に、NAT に構成できる設定とその説明を示します。

.

NAT 属性	説明	デフォルト値
NAT Address	BIG-IP 上のルーティング可能な外部 IP アドレスであり、外部ノードはこのアドレスに接続を送信したり（受信接続の場合）、このアドレスから接続を受信（発信接続の場合）したりします。	デフォルト値なし
Origin Address	内部ノードのプライベートクラス IP アドレスであり、外部ネットワークのノードはこのアドレスにアクセスします。	デフォルト値なし
State	NAT の状態で、[Enabled] または [Disabled] です。	**Enabled**
ARP	BIG-IP に、指定した NAT アドレスからの ARP 要求に応答し、ルータテーブル更新の Gratuitous ARP 要求を送信するよう指示する設定。	**Enabled**
VLAN Traffic	NAT Address または Origin Address に対応する特定の VLAN。複数の内部 VLAN がある場合など、NAT がマッピングされていない VLAN は、明示的に無効にできます。	**All VLANS**

***表 13.1** NAT の設定*

その他の制約事項

NAT を使用する場合、次の制約事項に注意してください。

- [**Origin Address**] ボックスで定義する IP アドレスは、内部ネットワーク上に実際に置かれた特定のサーバに対応している必要があります。
- NAT を再設定することはできず、いったん削除して作成しなおす必要があります。

NAT の管理

設定ユーティリティを利用して、既存の NAT をさまざまな方法で管理することができます。たとえば、標準 NAT が元の IP アドレスを変換アドレスにマッピングする方法を変更した方がいい場合もあります。

NAT の管理で実行できるタスクには、次のものがあります。

- NAT の確認 / 変更
- NAT の削除
- ロードバランシングプールでの NAT の有効化 / 無効化

◆注

NAT を管理するには、ユーザアカウントに適切なユーザロールが割り当てられている必要があります。

NAT の確認 / 変更

これまでに作成した NAT を確認または変更することができます。

NAT を表示または変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SNATs**] をクリックします。
   既存の SNAT のリストが表示されます。
2. メニューバーで、[**NAT List**] をクリックします。
   既存の NAT の一覧が表示されます。
3. [NAT] カラムで、NAT アドレスをクリックします。
   NAT のプロパティが表示されます。
4. 設定を変更し、[**Update**] をクリックします。

NAT の削除

これまでに作成した NAT を削除することができます。

NAT を削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SNATs**] をクリックします。
   既存の SNAT のリストが表示されます。
2. メニューバーで、[**NAT List**] をクリックします。
3. [NAT] カラムで、削除する NAT を探して、左側の [Select] ボックスをチェックします。
4. 画面の下部で [**Delete**] をクリックします。
   確認ボックスが表示されます。
5. [**Delete**] をクリックします。

ロードバランシングプールでの NAT の有効化 / 無効化

ロードバランシングプールを設定する場合、このプールを使用する接続で NAT 変換を無効にするように指定できます。デフォルトでは、この設定は有効になっています。詳細については、第 4 章「*ロードバランシングプールの設定*」を参照してください。

14

SNAT の設定

- セキュアネットワークアドレス変換の概要
- SNAT プールの作成
- SNAT の作成
- SNAT の管理
- SNAT の例

セキュアネットワークアドレス変換の概要

セキュアネットワークアドレス変換（***SNAT***）は、接続に含まれる送信元 IP アドレスを、ユーザが定義する BIG-IP の IP アドレスに変換します。宛先ノードはこの新しい送信元アドレスを宛先アドレスとして使用して、要求に応答します。

受信接続（つまり、クライアントノードで開始された接続）では、デフォルト経路が BIG-IP 経由になっていない場合でも、SNAT によってサーバノードが常にBIG-IP経由で応答を送信するようになります。SNAT により BIG-IP 経由で応答が送信されるようになることから、クライアント側から見れば自らが要求を送信したアドレスから応答が送られてくるように映り、その応答を受け入れるという仕組みになっています。

発信接続（つまり、サーバノードで開始された接続）では、サーバノードの内部 IP アドレスが外部ホストから見えないように隠ぺいされた状態のまま、サーバでそのホストへの接続が開始されます。

◆ 重要

*SNAT の実装に先立ち、第 13 章「**NAT の設定**」を読んでおくことをお勧めします。*

SNAT は NAT によく似ていますが、主に以下の点が異なります。

- NAT ではオリジナルアドレスと変換アドレスが 1 対 1 でマップされるの対して、SNAT では複数のオリジナルアドレスを 1 つの変換アドレスにマップできるようになっています。1 つの SNAT オブジェクトで、ネットワーク上のすべてのノードアドレスを 1 つのパブリック IP アドレスにマップすることも可能です。
- NAT を使用する場合、内部ノードのすべてのポートが解放されますが、SNAT では、デフォルトでサポートされるのは UDP と TCP のみとなっています。これもまた、SNAT の安全性を NAT にくらべて高くしている要素の 1 つです。
- BIG-IP は SNAT 接続を追跡し、SNAT とバーチャルサーバが同じパブリック IP アドレスを使用することを許可します。NAT 接続では BIG-IP による追跡は行われず、その点で SNAT とは異なります。
- 内部ノードのトラフィックが BIG-IP に到達する内部 VLAN に対して、NAT を明示的に有効にする必要があります。デフォルトでは、作成した SNAT はすべての VLAN に対して有効になります。

SNAT を作成するには、設定ユーティリティ、Traffic Management シェル（**tmsh**）、**bigpipe** ユーティリティのいずれかを使用します。また、作成する SNAT のタイプによっては、iRule を書くことも可能です。

tmsh の使い方については『***Traffic Management Shell (tmsh) Reference Guide***』を、**bigpipe** ユーティリティの使い方については『***Bigpipe Utility Reference Guide***』を、それぞれ参照してください。また iRules™ については、F5 Networks DevCentral サイト（**http://devcentral.f5.com**）および第 17 章「*iRule の記述*」を参照してください。

クライアントで開始された（受信）接続での SNAT の概要

一般的なクライアント - サーバ型のネットワーク構成では、BIG-IP の標準アドレス変換メカニズムによってサーバ応答が BIG-IP 経由でクライアントに返され、オリジナルの宛先 IP アドレス変換が逆に行われるようになっています。この一般的なネットワーク構成とは以下のようなものです。

- BIG-IP と同じサブネット上にサーバノードがある。
- サーバノードとは別のサブネット上にクライアントノードがある。
- BIG-IP がサーバサブネットのデフォルトゲートウェイとなっている。

しかし、ネットワーク構成によっては、BIG-IP の標準アドレス変換シーケンスだけでは、サーバ応答で必要なリターンパスを使用することが難しいケースがあります。このような例外的な構成としては以下のようなものがあります。

- **クライアントとサーバが同じネットワーク上にある場合**
  クライアントノードと同じネットワーク上にあるサーバノードへの要求のロードバランシングを行う場合、SNAT を作成することで、サーバ応答をサーバノードからクライアントノードに直接送信するのではなく、バーチャルサーバ経由で返信できるようになります。これを行わなければ、応答の送信元と要求の宛先が一致しなくなり、クライアントが応答の受け取りを拒否するといった問題が生じる場合があります。この SNAT 構成は「*バーチャルサーババウンスバック*」と呼ばれ、応答の送信元と要求の宛先が一致するようになることから、クライアントノードで応答が確実に受け入れられます。このような構成は、Web サーバから同じネットワーク上のアプリケーションサーバに要求の負荷を分散したい場合に使用することができます。
- **サーバノードのデフォルトゲートウェイが BIG-IP ではない場合**
  さまざまな理由から、サーバノードのデフォルト経路は必ずしも BIG-IP を経由したものになるわけではありません。この場合も、応答の送信元と要求の宛先が一致しないため、クライアントが応答の受け取りを拒否するといった問題が生じる可能性があります。問題を解決するには SNAT を作成します。BIG-IP は要求に含まれるクライアントノードの送信元 IP アドレスを SNAT アドレスに変換することから、サーバノードがその SNAT アドレスを宛先アドレスとして使用して、応答を送信するようになります。これによって、サーバノードのデフォルトゲートウェイではなく BIG-IP を経由して、強制的にクライアントノードに応答を返すことができます。
- **OneConnect 機能を使用している場合**
  BIG-IP の OneConnect 機能を使えば、アイドル状態のサーバ側接続をクライアントリクエストで再利用することができます。SNAT を使わなければ、どのクライアントノードが接続を再利用するかに関係なく、その接続を最初に確立したクライアントノードのアドレスがそのままサーバ側接続の発信元 IP アドレスとして残ります。これはトラフィックルーティングでは問題にならないものの、さまざまなシステム出力を確認する際に混乱を招くおそれがあります。SNAT を作成することでこの問題が解消されます。

◆注

受信接続でSNAT を使用すると、短命ポートのアベイラビリティに影響が出る可能性があります。その場合、いくつかの送信元ポートが利用可能になるまで、SNAT がそれ以上接続を処理できない状態になるかもしれません。

図 14.1 では、BIG-IP がサーバのデフォルトゲートウェイとして指定されておらず、かつ受信トラフィックで SNAT を設定していない場合に、クライアントで開始された接続に生じる典型的な問題について解説しています。

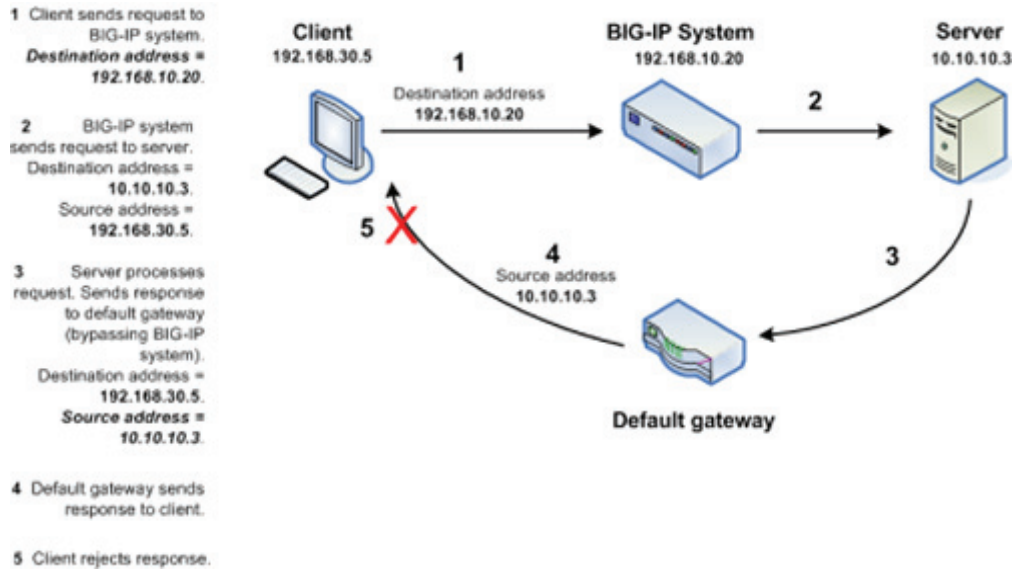

図 14.1 *宛先 IP アドレスと送信元 IP アドレスが一致しないためクライアントが応答の受け取りを拒否*

このような問題を防止するには、受信 SNAT を設定します。***受信SNAT*** は、要求に含まれるオリジナルのクライアント送信元 IP アドレスを、BIG-IP のバーチャルサーバもしくは BIG-IP 自身の IP アドレスに変換することで、その後のサーバ応答が BIG-IP に直接送信されるようにします。受信 SNAT が設定されている場合、BIG-IP は要求に含まれる宛先 IP アドレス（標準のアドレス変換メカニズムを使用）だけでなく、その要求の送信元 IP アドレス（SNAT を使用）もあわせて変換します。

図 14.2（14-4 ページ）を見れば、SNAT を設定することで、応答がデフォルトゲートウェイではなく BIG-IP 経由で送信され、クライアントがサーバ応答を受け入れられるようになっているのが分かります。

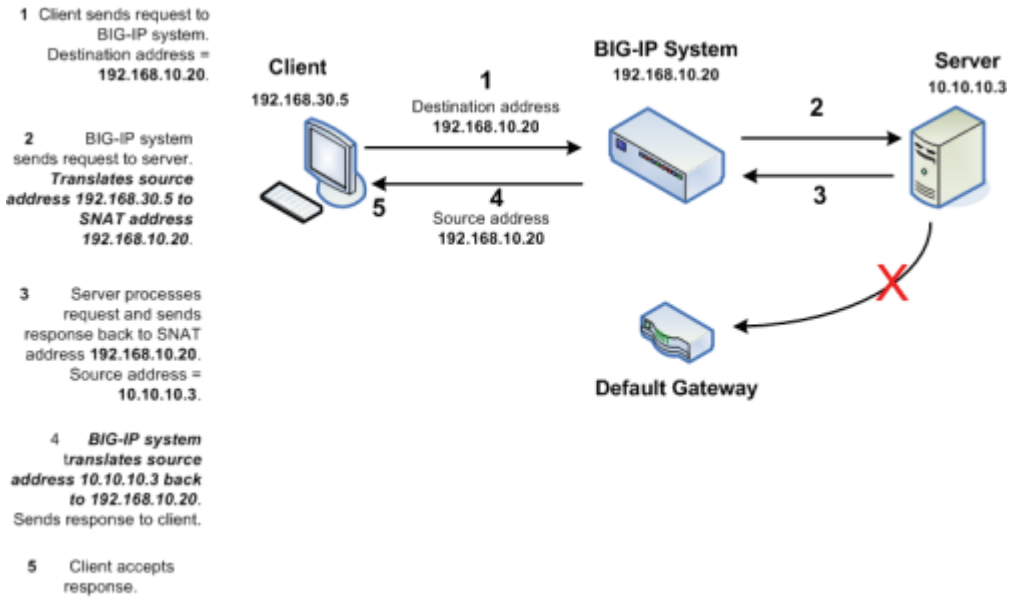

図 14.2 宛先 IP アドレスと送信元 IP アドレスが一致しているためクライアントが応答を受信

サーバで開始された（発信）接続での SNAT の概要

内部サーバから外部ホストへの接続が開始されると、SNAT は発信接続に含まれる 1 つ以上のサーバのプライベートの発信元 IP アドレスを、パブリックにルーティング可能な 1 つのアドレスに変換します。外部の宛先ホストはこのパブリックアドレスを宛先アドレスとして使用し、応答を送信します。この場合、内部ノードのプライベートクラス IP アドレスは外部ホストから見えないようになっています。

具体的には、発信接続での SNAT は以下のように機能します。

1. BIG-IP がオリジナル IP アドレス（つまり、プライベート IP アドレスを持つ内部サーバ）からパケットを受信し、その送信元アドレスが SNAT で定義されているかどうかを確認します。
2. オリジナル IP アドレスが SNAT で定義されている場合、BIG-IP はその送信元 IP アドレスを SNAT で定義された変換アドレスに変更します。
3. 続いて BIG-IP は、SNAT 変換アドレスを送信元アドレスとして使い、パケットを宛先ホストに送信します。

図 14.3 は発信 SNAT の例です。この例では、**172.16.20.4**、**172.16.20.5**、**172.16.20.6** という IP アドレスを持つ 3 台の内部ノードが、BIG-IP 経由で送信を行い、その結果、3 つの発信接続の中で **207.10.1.102** というパブリック IP アドレスが送信元 IP アドレスとして示されています。

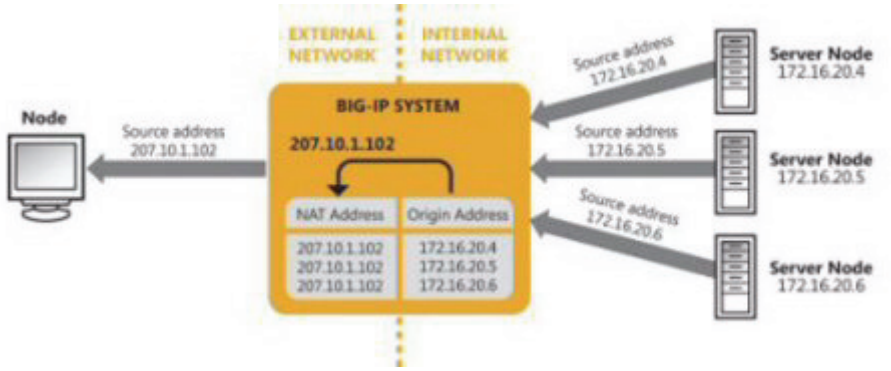

図 14.3 複数の発信接続での SNAT の使用例

SNAT 実装の概要

SNAT を作成したら、それぞれのニーズに応じて、以下のいずれかの方法でオリジナル IP アドレスを変換アドレスにマップします。

- 1 つ以上のオリジナル IP アドレスを、1 つの変換アドレスに明示的に割り当てる（図 14.3（14-5 ページ）を参照）。
- SNAT automap 機能を使用。
- 「***SNAT プール***」と呼ばれる変換アドレスプールを作成し、1 つ以上のオリジナル IP アドレスをその SNAT プールにマップする。
- SNAT プールを作成し、すべてのオリジナル IP アドレスをその SNAT プールにマップする。

オリジナル IP アドレスを 1 つの変換アドレスにマップ

SNAT を作成する方法の 1 つに、1 つ以上の元の IP アドレスを、選択した特定の変換アドレスに直接マッピングする方法があります。このタイプの SNAT を作成するには、設定ユーティリティの [New SNAT] 画面を使用します。このタイプの SNAT の実装方法については、「*SNAT の作成*」（14-9 ページ）を参照してください。

SNAT automap 機能の使用

SNAT automap という BIG-IP の機能を利用して SNAT を作成する方法もあります。***SNAT automap*** 機能では、BIG-IP 自身の IP アドレスからいずれか 1 つ（通常は出力 VLAN のフローティングセルフ IP アドレス）が自動的に選択され、オリジナル IP アドレスまたは SNAT 作成時に指定したアドレスにそれがマップされます。この機能を使用すると、変換アドレスを明示的に指定する必要がなくなります。

セルフ IP アドレスが自動で選択され、指定したオリジナル IP アドレスにマップされる際、静的な（非フローティング）セルフ IP アドレスよりもフローティングセルフ IP アドレスが優先されます。これによって、フェイルオーバ発生時にサービスが中断することがなくなります。

なお、出力 VLAN にフローティングセルフ IP アドレスが現在割り当てられていない場合は、非出力 VLAN のフローティング IP アドレスが使用されます。

このタイプの SNAT の実装方法については、「*SNAT の作成*」（14-9 ページ）を参照してください。

オリジナル IP アドレスを SNAT プールにマップ

変換アドレスプールを作成して、1 つ以上のオリジナル IP アドレスをその変換プール全体にマップすることで、SNAT を作成することもできます。この変換アドレスプールは、***SNAT プール***と呼ばれます。SNAT プールは、設定ユーティリティの [New SNAT Pool] 画面を使用して作成します。SNAT プールの作成方法の詳細については、「*SNAT の作成*」（14-9 ページ）を参照してください。

SNAT プールを作成して元の IP アドレスにマッピングすると、バーチャルサーバが元の IP アドレスからパケットを受信して、BIG-IP が SNAT プールから変換アドレスを選択します。BIG-IP は、元の IP アドレスを選択したアドレスに変換します。

元の IP アドレスを SNAT プールにマッピングするには、次の 2 つの方法があります。

- **SNAT オブジェクトを作成する**
  SNAT プールにアドレスをマップするには、設定ユーティリティの [New SNAT] 画面を使用します。詳細については、「*標準 SNAT の作成*」（14-9 ページ）を参照してください。
- **iRule を作成する**
  この場合、SNAT オブジェクトは作成されません。**snat** または **snatpool** コマンドを持つ iRule を作成します。iRule を記述して作成した SNAT の種類は、インテリジェント SNAT と呼ばれます。***インテリジェント SNAT*** では、1 つ以上の元の IP アドレスを、iRule を使用して変換アドレスにマッピングします。インテリジェント SNAT の詳細については、「*インテリジェント SNAT の作成*」（14-12 ページ）を参照してください。

すべての元の IP アドレスの変換アドレスプールへのマッピング

SNAT を作成するには、SNAT プールを作成して（設定ユーティリティの[New SNAT Pool]画面を使用）、バーチャルサーバにそのリソースとして直接割り当てるという方法もあります。SNAT プールをバーチャルサーバに割り当てると、BIG-IP はバーチャルサーバ経由で受け取るすべての元の IP アドレスを、その SNAT プールに自動的に割り当てます。インテリジェント SNAT と同様に、設定ユーティリティの [New SNAT] 画面では SNAT オブジェクトを作成しません。この種類の SNAT の詳細については、「*SNAT プールのバーチャルサーバへの直接割り当て*」（14-13 ページ）を参照してください。

SNAT プールの作成

SNAT プールを使用して SNAT で変換アドレスを指定する場合は、まず SNAT プールを作成してから、SNAT プールに追加する変換アドレスを 1 つ以上指定する必要があります。SNAT プールは、設定ユーティリティを使用して作成します。SNAT プールの背景情報については、「*オリジナル IP アドレスを SNAT プールにマップ*」（14-6 ページ）を参照してください。

SNAT プールの作成後、ニーズに最適な種類の SNAT を作成します（標準 SNAT、インテリジェント SNAT、またはバーチャルサーバに直接割り当てる SNAT プール）。作成できる SNAT の種類については、「*SNAT の作成*」（14-9 ページ）を参照してください。

SNAT プールには、作成時に構成する必要のある設定が 2 つあります。表 14.1 に、これらの設定を列挙して説明します。

プロパティ	説明	デフォルト値
Name	SNAT プールの一意の名前	デフォルト値なし
Member List	SNAT プールに含める IP アドレスのリスト。追加した IP アドレスがまだ変換アドレスとして指定されていない場合、BIG-IP が自動的に指定し、デフォルト値で適切なプロパティを割り当てます。この設定は必須です。	デフォルト値なし

表 14.1 *SNAT プールのプロパティ*

SNAT プールに追加した各変換アドレスの設定は、SNAT プールにアドレスを追加した後に設定できます。各設定の詳細については、「*変換アドレスの指定*」（14-11 ページ）を参照してください。

SNAT プールを作成した後、次のうちの 1 つを実行する必要があります。

- 作成した SNAT オブジェクト内から、SNAT プールを参照します。これは、標準 SNAT を作成した場合に実行します。詳細については、「*標準 SNAT の作成*」（14-9 ページ）を参照してください。
- iRule 内から SNAT プールを参照して、iRule をリソースとしてバーチャルサーバに割り当てます。これは、インテリジェント SNAT を作成した場合に実行します。詳細については、「*インテリジェント SNAT の作成*」（14-12 ページ）を参照してください。
- SNAT プールをリソースとしてバーチャルサーバに直接割り当てます。詳細については、「*SNAT プールのバーチャルサーバへの直接割り当て*」（14-13 ページ）を参照してください。

◆注

*SNAT プールをパーティション内に置くことはできません。したがって、ユーザが SNAT プールを作成・管理できるかどうかは、パーティションアクセスの割り当てではなく、ユーザロールによって決まります。詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

SNAT プールを作成するには

1. ナビゲーションペインの [Main] タブで [**Local Traffic**] を展開し、[**SNATs**] をクリックします。
   [SNATs] 画面が開きます。
2. メニューバーで、[**SNAT Pool List**] をクリックします。
   既存の SNAT プールのリストが表示されます。
3. 画面右上の [**Create**] をクリックします。

   ***注 : [Create]** ボタンをクリックできない場合は、現在のユーザロールではSNATを作成する権限が与えられていないことを示します。*
4. [**Name**] 設定には、SNAT プールの一意の名前を入力します。
5. [**Member list**] 設定に、IP アドレスを入力します。
6. [**Add**] をクリックします。
7. 追加する変換アドレスそれぞれに対して、ステップ 5 と 6 を繰り返します。
8. [**Finished**] をクリックします。

SNAT の作成

セキュアネットワークアドレス変換を実装する前に、作成する SNAT の種類を決めておく必要があります。次の種類の SNAT を作成できます。

- **標準 SNAT**
  標準 SNAT は、設定ユーティリティを使用して作成するオブジェクトで、1 つ以上の元の IP アドレスの変換アドレスへのマッピングを指定します。この種類の SNAT では、変換アドレスを適用する時期を BIG-IP が判断する基準は、完全に元の IP アドレスに基づいています。つまり、SNAT で指定した元の IP アドレスからパケットが到着すると、BIG-IP はそのアドレスを指定した変換アドレスに変換します。

  作成できる標準 SNAT には 3 種類あります。
  - 特定の変換アドレスに指定した SNAT
  - automap 機能を使用する SNAT
  - 変換アドレスとして SNAT プールを指定した SNAT

- **インテリジェント SNAT**
  標準 SNAT と同様に、***インテリジェント SNAT*** は、1 つ以上の元の IP アドレスの変換アドレスへのマッピングです。ただし、この種類の SNAT マッピングは、iRule 内で実装します。SNAT オブジェクトは作成しません。この種類の SNAT では、変換アドレスを適用する時期を BIG-IP が判断する基準は、HTTP Cookie またはサーバポートなどの、iRule 内で指定したデータの断片に基づいています。

- **バーチャルサーバリソースとして割り当てられた SNAT プール**
  この種類の SNAT は、バーチャルサーバにリソースとして直接割り当てた SNAT プールのみから構成されます。この種類の SNAT を実装するには、SNAT プールのみを作成します。SNAT オブジェクトや iRule を作成する必要はありません。

元の IP アドレスの変換アドレスへのマッピングについては、第 13 章「*NAT の設定*」を参照してください。

標準 SNAT の作成

標準 SNAT は、設定ユーティリティを使用して作成します。元の IP アドレスにマッピングする変換アドレスには、特定の IP アドレス、既存の SNAT プール、それ自身の IP アドレス（automap 機能を使用）を指定できます。

変換アドレス用の SNAT プールを使用する場合は、SNAT よりも先に SNAT プールを作成しておく必要があります。詳細については、「*SNAT プールの作成*」（14-7 ページ）を参照してください。

標準 SNAT を作成する場合、BIG-IP は SNAT に一連のプロパティを自動的に割り当てます。SNAT を作成する際に **Name** 設定と **Translation** 設定を行う必要がありますが、その他の設定にはデフォルト値を使用したり、後でその値を変更したりできます。

◆注

*SNAT をパーティション内に置くことはできません。したがって、ユーザが SNAT を作成・管理できるかどうかは、パーティションアクセスの割り当てではなく、ユーザロールによって決まります。詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

標準 SNAT を作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SNATs**] をクリックします。
   [SNATs] 画面が開きます。
2. 画面右上の [**Create**] をクリックします。
   ***注 : [Create]** ボタンをクリックできない場合は、現在のユーザロールではSNATを作成する権限が与えられていないことを示します。*
3. [**Name**] 設定には、SNAT の一意の名前を入力します。
4. [**Translation**] 設定では、[**IP Address**]、[**SNAT Pool**]、または [**Automap**] を選択します。
5. [**IP Address**] または [**SNAT Pool**] を選択した場合、IP アドレスを入力するか、SNAT プール名を選択します。
6. その他すべての値を、変更するか、そのまま使用します。
7. [**Finished**] をクリックします。

表 14.2 に、SNAT に構成できる設定を示します。表に続けて、各設定の詳細について説明します。

プロパティ	説明	デフォルト値
Name	標準 SNAT の一意の名前を指定します。このプロパティの設定は必須です。	デフォルト値なし
Translation	選択した値に従って、個々の IP アドレス、SNAT プール名、Automap オプションを指定します。指定できる値は、**IP Address**、**SNAT Pool**、または **Automap** です。	**Automap**
Origin	変換アドレス、あるいは変換アドレスまたはセルフ IP アドレスのプールをマッピングする先の、元の IP アドレスを指定します。指定できる値は、**All Addressess**、または **Address List** です。	**All Addresses**
VLAN Traffic	SNAT を適用する対象の VLAN です。指定できる値は、**All VLANS**、**Enabled On**、および **Disabled On** です。	**ALL VLANS**

***表 14.2** 標準 SNAT のプロパティ*

SNAT 名の指定

SNAT で設定する、最も基本的な設定は SNAT 名です。SNAT 名では大文字と小文字が区別され、文字、数字、下線（_）のみを使用できます。予約済みキーワードは使用できません。

定義する各 SNAT には、一意の名前が必要です。

変換アドレスの指定

[**Translation**] 設定では、元の IP アドレスにマッピングする変換アドレスを指定します。変換アドレスの背景情報については、「*セキュアネットワークアドレス変換の概要*」（14-1 ページ）を参照してください。

[**Translation**] 設定に指定できる値には、3 種類あります。

- **IP Address**
  SNAT を作成する場合、SNAT で変換アドレスとして使用する IP アドレスを指定できます。
- **SNAT pool**
  この値を指定すると、既存の SNAT プールを、元の IP アドレスのマッピング先として指定できます。SNAT プールの詳細と作成方法については、「*SNAT プールの作成*」（14-7 ページ）を参照してください。SNAT プールを使用する標準 SNAT の例については、「*例 1 - SNAT プールを使用する標準 SNAT の確立*」（14-18 ページ）を参照してください。
- **Automap**
  SNAT プールと同様に、SNAT automap 機能では、1 つ以上の元の IP アドレスを、変換アドレスのプールにマッピングできます。ただし、SNAT automap 機能では、プールを作成する必要はありません。BIG-IP が、すべてのセルフ IP アドレスをプールの変換アドレスとして使用して、プールを効率的に作成します。

変換アドレスまたは SNAT プールを指定する場合、BIG-IP は変換アドレスに一連のプロパティを自動的に割り当てます。これらのプロパティにデフォルト値を使用することも、必要に応じて変更することもできます。表 14.3 に、変換アドレスのプロパティを列挙して説明します。

プロパティ	説明	デフォルト値
IP address	変換アドレスとして指定する IP アドレス。この設定は必須です。	デフォルト値なし
State	変換アドレスの状態で、[enabled] または [disabled] です。[**disabled**] に設定されていると、変換アドレスは接続開始には使用されません。	**Enabled**
ARP	BIG-IP が ARP 要求に応答するか、または Gratuitous ARP を送信するかを決定する設定。	**Enabled**
Connection Limit	変換アドレスが接続を開始しなくなるまでの、接続数の制限。デフォルト値 **0** は、設定が無効であることを示します。	**0**

***表 14.3** SNAT 変換アドレスのプロパティ*

プロパティ	説明	デフォルト値
TCP Idle Timeout	SNATアドレスを使用して開始されたTCP接続が自動的に切断されるまでの、アイドル状態になる秒数を定義するタイマ。指定できる値は、[**Indefinite**]、または [**Specify**] です。	**Indefinite**
UDP Idle Timeout	SNAT アドレスを使用して開始された UDP 接続が自動的に切断されるまでの、アイドル状態になる秒数を定義するタイマ。指定できる値は、[**Indefinite**]、または [**Specify**] です。	**Indefinite**
IP Idle Timeout	SNAT アドレスを使用して開始された IP 接続が自動的に切断されるまでの、アイドル状態になる秒数を定義するタイマ。指定できる値は、[**Indefinite**]、または [**Specify**] です。	**Indefinite**

***表 14.3** SNAT 変換アドレスのプロパティ（続き）*

変換アドレスの定義、表示、変更については、「*SNAT の管理*」（14-14 ページ）を参照してください。

元の IP アドレスの指定

[**Origin**] 設定では、変換アドレスにマッピングする元の IP アドレスを指定します。この設定には、値として 1 つの IP アドレス、または複数の IP アドレスを追加できます。

VLAN トラフィックの指定

[**VLAN Traffic**] 設定では、SNAT を適用する VLAN を指定します。指定できる値は、**All VLANS**、**Enabled On**、および **Disabled On** です。

インテリジェント SNAT の作成

セキュアアドレス変換を実行する方法の 1 つに、インテリジェント SNAT を作成する方法があります。前述したように、***インテリジェント SNAT*** は、SNAT オブジェクトではなく、1 つ以上の元の IP アドレスを変換アドレスへマッピングする iRule です。インテリジェント SNAT を作成するには、次のタスクを実行する必要があります。

- 元の IP アドレスを SNAT プール（個別の変換アドレスではなく）にマッピングするには、[New SNAT Pools] 画面を使用して、これらの変換アドレスをメンバとして含む 1 つ以上の SNAT プールを作成します。詳細については、「*SNAT プールを作成するには*」（14-8 ページ）を参照してください。
- [New iRules] 画面を使用して、**snat** または **snatpool** コマンドを持つ iRule を作成します。iRule コマンドは、BIG-IP が変換アドレスの選択に使用する、変換アドレスまたは変換アドレスのプールを指定します。iRules の詳細については、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** または第 17 章「*iRule の記述*」を参照してください。

- 該当するバーチャルサーバの [Resources] 画面から、iRule をリソースとしてバーチャルサーバに割り当てます。バーチャルサーバの詳細については、第 2 章「*バーチャルサーバの設定*」を参照してください。

◆注

*インテリジェント SNAT の例については、「**例 2 - インテリジェント SNAT の確立**」（14-19 ページ）を参照してください。*

SNAT プールのバーチャルサーバへの直接割り当て

SNAT オブジェクトを作成したり、iRule を使用してインテリジェント SNAT を作成するのではなく、SNAT プールを作成して、リソースとしてバーチャルサーバに直接割り当てることもできます。これによって、変換アドレスをマッピングする元の IP アドレスを明示的に定義する必要がなくなります。

SNAT プールの作成方法の詳細については、「*SNAT プールの作成*」（14-7 ページ）を参照してください。SNAT プールのバーチャルサーバへの直接割り当ての詳細については、第 2 章「*バーチャルサーバの設定*」を参照してください。

SNAT の管理

設定ユーティリティを使用して、既存の SNAT をさまざまな方法で管理できます。たとえば、新しい SNAT プールを作成する前に、既存のプールのリストを表示したい場合があります。または、標準 SNAT が元の IP アドレスを変換アドレスにマッピングする方法を変更したい場合があります。

SNAT を管理する際に実行できるタスクには、次のものがあります。

- SNAT または SNAT プールの表示または変更
- 既存の SNAT の変換アドレスの定義
- 既存の SNAT の変換アドレスの表示または変更
- SNAT、SNAT プール、変換アドレスの削除
- ロードバランシングプールでの SNAT の有効化または無効化
- SNAT 変換アドレスの有効化または無効化

SNAT、SNAT プールの表示または変更

これまでに作成した SNAT、NAT、SNAT プールを表示または変更できます。

SNAT を表示または変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SNATs**] をクリックします。
   既存の SNAT のリストが表示されます。
2. [SNAT] カラムで、SNAT 名をクリックします。
   SNAT のプロパティが表示されます。
3. 表示された設定を表示または変更します。
4. 設定を変更した場合、[**Update**] をクリックします。

SNAT プールを表示または変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SNATs**] をクリックします。
   既存の SNAT のリストが表示されます。
2. メニューバーで、[**SNAT Pool List**] をクリックします。
   既存の SNAT プールのリストが表示されます。
3. SNAT プール名をクリックします。
4. 表示された設定を表示または変更します。
5. 設定を変更した場合、[**Update**] をクリックします。

変換アドレスの定義または表示

変換アドレスを定義できるほか、前に定義した既存の変換アドレスを表示・変更することも可能です。

既存の SNAT の変換アドレスを明示的に定義するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SNATs**] をクリックします。
2. メニューバーで、[**SNAT Translation List**] をクリックします。既存の変換アドレスが表示されます。
3. 画面右上の [**Create**] をクリックします。

   注 : [Create] *ボタンをクリックできない場合は、現在のユーザロールではSNATを作成する権限が与えられていないことを示します。*
4. すべてのプロパティ設定を、そのまま使用するか、または変更します。
5. [**Finished**] をクリックします。

既存の SNAT の変換アドレスを表示または変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SNATs**] をクリックします。
   既存の SNAT のリストが表示されます。
2. メニューバーで、[**SNAT Translation List**] をクリックします。既存の変換アドレスのリストが表示されます。
3. 変換アドレスをクリックします。
4. 表示された設定を表示または変更します。
5. 設定を変更した場合、[**Update**] をクリックします。

SNAT、SNAT プール、変換アドレスの削除

これまでに作成した、既存の SNAT、SNAT プール、または変換アドレスを削除できます。

SNAT を削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SNATs**] をクリックします。
   既存の SNAT のリストが表示されます。
2. [SNAT] カラムで、削除する SNAT を探して、左側の [Select] ボックスをチェックします。
3. 画面の下部で [**Delete**] をクリックします。
   確認ボックスが表示されます。
4. [**Delete**] をクリックします。

SNAT プールを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SNATs**] をクリックします。
   既存の SNAT のリストが表示されます。
2. メニューバーで、[**SNAT Pool List**] をクリックします。
   既存の SNAT プールのリストが表示されます。
3. 削除する SNAT プールを探して、左側の [Select] ボックスをチェックします。
4. 画面下部の [**Delete**] をクリックします。
   確認ボックスが表示されます。
5. [**Delete**] をクリックします。

変換アドレスを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SNATs**] をクリックします。
   既存の SNAT のリストが表示されます。
2. メニューバーで、[**SNAT Translation List**] をクリックします。
   既存の変換アドレスのリストが表示されます。
3. 削除する変換アドレスを探して、左側の [Select] ボックスをチェックします。
4. 画面下部の [**Delete**] をクリックします。
   確認ボックスが表示されます。
5. [**Delete**] をクリックします。

ロードバランシングプールでの SNAT または NAT の有効化または無効化

ロードバランシングプールを設定する場合、このプールを使用する接続で SNAT 変換を無効にするように指定できます。デフォルトでは、この設定は有効になっています。詳細については、第 4 章「*ロードバランシングプールの設定*」を参照してください。

SNAT 変換アドレスの有効化または無効化

設定ユーティリティを使用して、個別の SNAT 変換アドレスを有効または無効にできます。

SNAT 変換アドレスを有効または無効にするには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**SNATs**] をクリックします。
2. メニューバーで、[**SNAT Translation List**] をクリックします。
3. 有効または無効にする変換アドレスを探して、左側の [Select] ボックスをチェックします。
4. 画面下部の [**Enable**] または [**Disable**] をクリックします。

SNAT の例

次の例では、SNAT プールを使用する SNAT の実装方法を示します。例では、以下の実行方法を説明します。

- SNAT プールを使用する標準 SNAT の確立
- インテリジェント SNAT の確立

◆注

*次の例では、SNAT プールを使用する SNAT をわかりやすく説明するため、BIG-IP の **bigip.conf** ファイルのサンプルエントリを示しています。**bigip.conf** ファイルのエントリは、設定ユーティリティを使用して BIG-IP を設定した結果を表しています。*

例 1 - SNAT プールを使用する標準 SNAT の確立

元の IP アドレスを個別の変換アドレスではなく、SNAT プールにマッピングする SNAT を作成する必要がある場合があります。このタイプの SNAT について説明するために、仮にインターネットサービスプロバイダ（ISP）が BIG-IP を所有していて、2 人の顧客にインターネットへのリンクとして、転送可能な IP アドレスを 2 つずつ付与するとします。顧客はこれらの転送可能 IP アドレスを、顧客自身のサーバへの受信トラフィックの仮想 IP アドレスとして、また自身のサーバからの送信トラフィックの変換アドレスとして使用する必要があります。

この場合、SNAT を使用して解決できます。SNAT を実装するため、ISP は次の 3 つの手順を実行します。

まず、ISP は、図 14.4 に示す、ロードバランシングプール **isp_pool** を作成します。

```
pool isp_pool {
lb_method rr
member 199.5.6.254:0
member 207.8.9.254:0
}
```

***図 14.4** 基本ロードバランシングプールの **bigip.conf** エントリ*

次に ISP は、**customer1_snatpool**、**customer2_snatpool**、**other_snatpool** の 3 つの SNAT プールを作成します。図 14.5（14-19 ページ）にこれを示します。BIG-IP は、SNAT プールメンバを変換アドレスとして自動的に指定します。

```
snatpool customer1_snatpool {
member 199.5.6.10
member 207.8.9.10
}
snatpool customer2_snatpool {
member 199.5.6.20
member 207.8.9.20
}
snatpool other_snatpool {
member 199.5.6.30
member 207.8.9.30
}
```

***図 14.5** 3 つの SNAT プールの **bigip.conf** エントリ*

最後に、設定ユーティリティを使用して、ISP は元の IP アドレスをそれぞれ適切なSNATプールに直接マッピングするSNATを作成します。

例 2 - インテリジェント SNAT の確立

SNAT マッピングを元の IP アドレスではなく、サーバポート などを基準とするには、iRule を記述して、iRule 内に SNAT プールを指定します。この場合、設定ユーティリティの [SNAT] 画面を使用して、SNAT プールのみを作成し、実際の SNAT オブジェクト は作成しません。

たとえば、ISP などのユーザがインターネットへの 2 つの冗長接続を維持しているとします。また、ISP は多数の同時 CHAT 接続（ポート **531** を使用）を処理しており、サーバ側クライアントポートを使い切らないようにしたいとします。さらに、ISP は CHAT、SMTP、およびその他すべてのトラフィックごとに統計を収集したいとします。この場合、インテリジェント SNAT を設定することが、変換アドレスを選択する最適な方法です。

インテリジェント SNAT を実装するため、ISP は次の手順を実行します。

まず、**out_pool** というロードバランシングプールを作成します。**bigip.conf** ファイルでは、プールが図 14.6 のサンプルのようになります。

```
pool out_pool {
lb_method round_robin
member 199.5.6.254:0
member 207.8.9.254:0
}
```

***図 14.6** インテリジェント SNAT で使用されるプールの **bigip.conf** エントリ*

次に、図 14.7 に示すように、ISP は、設定ユーティリティを使用して、**199.5.6.10**、**199.5.6.11**、**207.8.9.10**、**207.8.9.11** の 4 つの IP アドレスを含む、**chat_snatpool** という SNAT プールを作成します。

BIG-IP は、SNAT プール作成中に、これらの IP アドレスを変換アドレスとして自動的に指定します。これらのアドレスは、CHAT トラフィックで使用される、次の 2 つのホップネットワークに対応します。**bigip.conf** ファイルでは、SNAT プールは図 14.7 のサンプルのようになります。

```
snatpool chat_snatpool {
member 199.5.6.10
member 199.5.6.11
member 207.8.9.10
member 207.8.9.11
}
```

***図 14.7** CHAT トラフィック用の SNAT プール定義*

次に、各変換アドレスに対して、ISP は設定ユーティリティを使用して TCP 接続のタイムアウト値を **600** に変更します。

次に ISP は、2 つの変換アドレス、**199.5.6.20** と **207.8.9.20** を含む 2 番目の SNAT プール、**smtp_snatpool** を作成します。これらのアドレスはそれぞれ、SMTP トラフィックで使用される、次の 2 つのホップネットワークのいずれかに対応します。**bigip.conf** ファイルでは、SNAT プールが図 14.8 のサンプルのようになります。

```
snatpool smtp_snatpool {
member 199.5.6.20
member 207.8.9.20
}
```

***図 14.8** SMTP トラフィック用の SNAT プール定義*

次に、ISP はその他すべてのトラフィック（CHAT や SMTP 以外のトラフィック）の SNAT プール、**other_snatpool** を作成します。ここでは、各 IP アドレスは、その他すべてのトラフィックで使用される、次の 2 つのホップネットワークのいずれかに対応します。図 14.9 にこれを示します。

```
snatpool other_snatpool { \SNAT pool definition
member 199.5.6.30
member 207.8.9.30
}
```

***図 14.9** その他すべてのトラフィック用の SNAT プール定義*

ISP は、開始パケットのサーバポートに基づく SNAT プールと、ロードバランシングプール**out_pool**の両方を選択するiRuleを記述します。

図 14.10 に、iRule でコマンド **TCP::local_port** を指定して、変換アドレスの選択基準として使用するパケットデータの種類を指定する方法を示します。また、iRule には、コマンド **snatpool** によって、BIG-IP が変換アドレスを選択する SNAT プールを指定する方法も示しています。

```
rule my_iRule {
when SERVER_CONNECTED
  if ( TCP::local_port equals 531 ) {
  snatpool chat_snatpool
}
  else if ( TCP::local_port equals 25 ) {
  snatpool smtp_snatpool
}
  else {
  snatpool other_snatpool
}
pool out_pool
}
```

***図 14.10** インテリジェント SNAT を参照する iRule の例*

iRule の **if** ステートメントでは、BIG-IP はクライアントリクエストのヘッダで指定されたサーバポートの値をテストします。結果に基づいて、BIG-IP は SNAT プールとロードバランシングプールの両方を選択します。

最終手順では、図 14.11 に示すように、ISP は iRule をリソースとしてワイルドカードバーチャルサーバに割り当てます。

```
virtual 0.0.0.0:0 use rule my_iRule
```

***図 14.11** iRule のワイルドカードバーチャルサーバへの割り当て*

15

トラフィッククラスの設定

- トラフィッククラスの概要
- トラフィッククラスの作成
- トラフィッククラスの設定
- トラフィッククラスの管理

トラフィッククラスの概要

BIG-IP® ローカルトラフィック管理システムには、トラフィッククラスと呼ばれる機能が搭載されています。トラフィッククラスは、WAN Optimization Module などのモジュールに最適化プロファイルを実装する際に使用する機能です。

トラフィッククラスを使えば、ユーザ定義の基準（送信元 IP アドレスや宛先 IP アドレスなど）に従い、トラフィックを分類することができます。トラフィッククラスの作成時には、分類基準だけでなく、分類 ID もあわせて定義します。トラフィッククラスを定義し、そのクラスをバーチャルサーバに割り当てると、BIG-IP によって***分類 ID***が各トラフィックフローに関連付けられます。この場合、BIG-IP はその分類に基づきトラフィックフローを調整することができます。

トラフィックフローとトラフィッククラスの照合の際には、可能なかぎり限定された照合が行われます。

トラフィッククラスを設定するには、BIG-IP の設定ユーティリティを使ってトラフィッククラスを作成し、そのクラスをバーチャルサーバに割り当てます。

◆注

***bigpipe** ユーティリティまたは**tmsh** を使ってトラフィッククラスを設定することも可能です。詳細については、**『Bigpipe Utility Reference Guide』**または**『Traffic Management Shell (tmsh) Reference Guide』**を参照してください。*

トラフィッククラスの作成

ほとんどのローカルトラフィックオブジェクトと同じように、トラフィッククラスも常にパーティション内に置かれます。したがって、トラフィッククラスを作成する前に、トラフィッククラスを配置するパーティションに対して現在の管理パーティションを設定しておく必要があります。管理パーティションとなるのは、そのトラフィッククラスを参照するバーチャルサーバが含まれるパーティションか、もしくは **Common** パーティションです。

たとえば、トラフィッククラスを参照するバーチャルサーバがパーティション **A** に置かれている場合、作成したトラフィッククラスをパーティション **A** もしくは **Common** パーティションに配置することができます。

トラフィッククラスを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して、[**Traffic Class**] をクリックします。
   [Traffic Class] 画面が開きます。
2. 画面右上の [**Create**] をクリックします。
3. 必要な設定すべてを設定します。
   設定の詳細については、「*トラフィッククラスの設定*」（15-2 ページ）、またはオンラインヘルプを参照してください。

4. [**Finished**] をクリックします。

トラフィッククラスを作成したら、バーチャルサーバにそれを割り当てる必要があります。バーチャルサーバについては、第 2 章「*バーチャルサーバの設定*」を参照してください。

トラフィッククラスの設定

表 15.1 で、トラフィッククラスで設定可能な項目について説明しています。

設定	説明	デフォルト値
Name	トラフィッククラス一意の名前を指定します。トラフィッククラス名では大文字と小文字が区別され、文字、数字、下線（_）のみを使用できます。予約済みキーワードは使用できません。この設定は必須です。	デフォルト値なし
Classification	トラフィッククラス基準にマッチするデータフローに関連付けるテキスト文字列を指定します。	デフォルト値なし
Source Address	そのクラスに属するトラフィックを識別するための送信元 IP アドレスを指定します。この IP アドレスから発信されるあらゆるトラフィックが、定義されたレートクラスの対象となります。	デフォルト値なし
Source Mask	[**Source Address**]設定で指定したアドレスのネットワークマスクを指定します。	デフォルト値なし
Source Port	そのクラスに属するトラフィックを識別するためのポート名もしくはポート番号を指定します。このポートから発信されるあらゆるトラフィックが、定義されたレートクラスの対象となります。	デフォルト値なし
Destination Address	そのクラスに属するトラフィックを識別するための宛先 IP アドレスを指定します。この IP アドレスに送信されるあらゆるトラフィックが、定義されたレートクラスの対象となります。	デフォルト値なし
Destination Mask	[**Destination Address**] 設定で指定したアドレスのネットワークマスクを指定します。	デフォルト値なし
Destination Port	そのクラスに属するトラフィックを識別するためのポート名もしくはポート番号を指定します。このポートに送信されるあらゆるトラフィックが、定義されたレートクラスの対象となります。	デフォルト値なし
IP Protocol	トラフィック分類の基準として使用する IP プロトコルを指定します。指定可能は値は、**TCP** と **UDP** です。たとえば、**TCP** と指定した場合、TCP 経由で BIG-IP に送信されるトラフィックがそのクラスのメンバとなります。	**TCP**

***表 15.1** トラフィッククラスの設定項目*

トラフィッククラスの管理

トラフィッククラスの作成後は、既存のトラフィッククラスの一覧を表示したり、トラフィッククラスの設定の確認・変更や、トラフィッククラスの削除を行うことができます。

既存のトラフィッククラスの一覧を表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して、[**Traffic Class**] をクリックします。
   既存のトラフィッククラスの一覧とその設定値が表示されます。
2. トラフィッククラスの一覧を確認します。

トラフィッククラスを確認・変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して、[**Traffic Class**] をクリックします。
   既存のトラフィッククラスの一覧が表示されます。
2. 一覧のトラフィッククラス名をクリックします。
   そのトラフィッククラスの設定が表示されます。
3. 設定値を、そのまま使用するか、または変更します。トラフィッククラスの設定については、「*トラフィッククラスの設定*」（15-2 ページ）を参照してください。
4. [**Update**] をクリックします。

トラフィッククラスを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して、[**Traffic Class**] をクリックします。
   既存のトラフィッククラスの一覧が表示されます。
2. 一覧からトラフィッククラス名を選び、名前の左側の [Select] ボックスをチェックします。
3. 画面の下部で [**Delete**] をクリックします。
   削除を確認する画面が表示されます。
4. [**Delete**] をクリックします。
   そのトラフィッククラスが削除されます。

16

レートシェーピングの設定

- レートシェーピングの概要
- レートクラスの作成と実装
- レートクラス設定の設定
- レートクラスの管理

レートシェーピングの概要

BIG-IP® ローカルトラフィックマネージメントシステムには、レートシェーピングと呼ばれる機能が搭載されています。*レートシェーピング*によって、受信トラフィックに対してスループットポリシーを適用できます。スループットポリシーは、選択したトラフィックパターンに対して帯域幅の優先順位を付けたり帯域幅を制限したりする場合に有効です。

レートシェーピングは、優先クライアントがある e コマースサイトにとって便利です。たとえば、優先顧客には高いスループットを設定し、その他のサイトトラフィックには低いスループットを設定したい場合があります。

レートシェーピング機能では、まず選択したパケットをレートクラスのキューに入れ、レートクラスで指定された順番に、指定されたレートでパケットをキューから取り出します。*レートクラス*は、レートクラスによって処理されるすべてのトラフィックに適用されるスループット制限とパケットスケジューリング手法を定義する、レートシェーピングポリシーです。

レートシェーピングは、1 つ以上のレートクラスを作成して、そのレートクラスをパケットフィルタまたはバーチャルサーバに割り当てることで設定できます。iRules™ 機能を使用して、BIG-IP でレートクラスを特定の接続に適用させることもできます。

レートクラスをサーバからクライアントへの、またはクライアントからサーバへのトラフィックに特定して適用できます。レートクラスをクライアントへのトラフィックに設定すると、BIG-IP はスループットポリシーをサーバに向かうトラフィックには適用しません。逆に、レートクラスをサーバへのトラフィックに設定すると、BIG-IP はスループットポリシーをクライアントに向かうトラフィックには適用しません。

レートシェーピングを設定するには、設定ユーティリティの [Local Traffic] セクションの [Rate Shaping] 画面を使用します。

レートクラスの作成と実装

レートクラスでは、レートクラスで処理するすべてのトラフィックに対して BIG-IP が適用する、スループット制限とパケットスケジューリング手法を定義します。レートクラスは、バーチャルサーバおよびパケットフィルタルールに割り当てられ、iRules 経由でも割り当てられます。

同じトラフィックに対して、複数の場所から割り当てられたレートクラスが適用されている場合、BIG-IP は最後に割り当てられたレートクラスのみを適用します。BIG-IP は、次の順番でレートクラスを適用します。

- BIG-IP が割り当てる最初のレートクラスは、トラフィックと指定したレートクラスとを照合させた、最後のパケットフィルタルールからのものです。
- 次に BIG-IP が割り当てるレートクラスは、バーチャルサーバがレートクラスを指定する場合はバーチャルサーバからのもので、このレートクラスはパケットフィルタが選択するレートクラスをオーバーライドします。
- 最後に割り当てられるレートクラスは、iRule がレートクラスを指定する場合の iRule からのもので、このレートクラスはこれまでに選択されたレートクラスをオーバーライドします。

◆注

*レートクラスをパーティション内に置くことはできません。したがって、ユーザがレートクラスを作成・管理できるかどうかは、パーティションアクセスの割り当てではなく、ユーザロールによって決まります。詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

レートクラスを作成するには、BIG-IP の設定ユーティリティを使用します。

◆注

***bigpipe** ユーティリティまたは**tmsh** を使ってレートクラスを設定することも可能です。詳細については、『**Bigpipe Utility Reference Guide**』または『**Traffic Management Shell (tmsh) Reference Guide**』を参照してください。*

レートクラスを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Rate Shaping**] をクリックします。
   [Rate Classes] 画面が開きます。
2. 画面右上の [**Create**] をクリックします。
   [New Rate Class] 画面が表示されます。
3. レートクラスが、親レートクラスから帯域幅を借用できるように設定するかどうかを指定します。
   - レートクラスが、親レートクラスから帯域幅を借用できるようにしない場合は、[**Basic**] を選択します。詳細については、「*帯域幅の借用*」（16-8 ページ）を参照してください。

- レートクラスが、親レートクラスから帯域幅を借用できるようにする場合は、[**Advanced**] を選択します。詳細については、「*親クラスの指定*」(16-8 ページ) を参照してください。

4. 必要な設定すべてを設定します。
   設定の詳細については、「*レートクラス設定の設定*」(16-4 ページ)、またはオンラインヘルプを参照してください。
5. [**Finished**] をクリックします。

レートクラスを作成した後、これをバーチャルサーバまたはパケットフィルタルールに割り当てる必要があります。または、iRule 内からレートクラスを指定します。

- バーチャルサーバの詳細については、第 2 章「*バーチャルサーバの設定*」を参照してください。
- パケットフィルタルールの詳細については、設定ユーティリティの [Packet Filter] 画面からオンラインヘルプを参照してください。
- iRules の詳細については、第 17 章「*iRule の記述*」を参照してください。

レートクラス設定の設定

レートクラスを作成する場合、BIG-IP はレートクラスにいくつかのデフォルト設定を割り当てます。これらのデフォルト値をそのまま使用することも、必要に応じて変更することもできます。レートクラスで設定できる項目について、表 16.1 に示します。

設定	説明	デフォルト値
Name	レートクラスの一意の名前を指定します。各レートクラスには名前が必要です。	デフォルト値なし
Base Rate	レートクラスが処理するトラフィックで許容される、ベーススループットを指定します。一般に、指定したレートをパケットが超過することはできません。この設定は必須です。	デフォルト値なし
Ceiling Rate	ベースレートと同様ですが、変更できない絶対的な制限を指定します。この数値は、トラフィックがバーストまたは借用する場合にフローできるレートの絶対制限値を指定します。帯域幅バーストおよび借用の詳細については、「*バーストサイズの指定*」（16-6 ページ）を参照してください。	[Base Rate] と同じ
Burst Size	トラフィックが帯域幅の借用が必要となるまで、ベースレートを超過してバーストできる最大バイト数を指定します。この値が **0** に設定されている場合、バーストはできません。帯域幅バーストおよび借用の詳細については、「*バーストサイズの指定*」（16-6 ページ）を参照してください。	**0**
Direction	レートクラスが適用されるトラフィックの方向を指定します。 **Any**: 両方向のトラフィックにレートクラスを適用します。 **Client**: サーバからクライアントに送信されるトラフィックにレートクラスを適用します。 **Server**: クライアントからサーバに送信されるトラフィックにレートクラスを適用します。	**Any**
Parent Class	このクラスが帯域幅を借用できるレートクラスを指定します。子レートクラスは、親レートクラスから未使用の帯域幅を借用でき、子レートクラスのバーストサイズを補強できます。これは **[Advanced]** 設定です。帯域幅バーストおよび借用の詳細については、「*バーストサイズの指定*」（16-6 ページ）を参照してください。	**None**
Shaping Policy	破棄ポリシーとキュー方式のカスタマイズ値を含むシェーピングポリシーを指定します。	**None**
Queue Method	レートクラスがトラフィックのキューおよびデキューに使用する方法を指定します。指定可能な設定は **pfifo** と **sfq** です。	親クラスが指定されている場合は親クラスと同じです。それ以外の場合は **sfq** となります。
Drop Policy	キューのトラフィックが一杯になった場合、必要に応じて、処理パケットをいつどのように破棄するかを指定します。	**tail**

***表 16.1** レートクラス設定の設定項目*

レートクラスの設定をする前に、これらの設定の説明を参照しておくと有益です。

名前の指定

レートクラスで最初に設定する項目は、レートクラス名です。レートクラス名では大文字と小文字が区別され、文字、数字、下線（_）のみを使用できます。予約済みキーワードは使用できません。

定義する各レートクラスには、一意の名前が必要です。この設定は必須です。

レートクラス名を指定するには、[New Rate Class] 画面の [**Name**] ボックスで、レートクラスの一意の名前を入力します。

ベースレートの指定

[**Base Rate**] 設定では、レートクラスが処理するトラフィックで許容される、ベーススループットを指定します。親レートクラスに付随するすべての子レートクラスのベースレートの合計に、親レートクラスのベースレートを足したものが、親レートクラスの上限を超えることがあってはなりません。そのため F5 では、親レートクラスのベースレートを常に **0**（デフォルト値）に設定することを推奨しています。

ベースレートは、ビット毎秒（bps）、キロビット毎秒（Kbps）、メガビット毎秒（Mbps）、またはギガビット毎秒（Gbps）で指定できます。デフォルトの単位はビット毎秒です。この設定は必須です。

◆注

この数値は、10 のべき乗で、2 のべき乗ではありません。

上限の指定

[**Ceiling Rate**] 設定は、トラフィックがバーストまたは借用する場合にフローできるレートの絶対制限値を指定します。上限は、ビット毎秒（bps）、キロビット毎秒（Kbps）、メガビット毎秒（Mbps）、またはギガビット毎秒（Gbps）で指定できます。デフォルトの単位はビット毎秒です。

レートクラスが親レートクラスの場合、この上限値によって、親レートクラスに付随するすべての子レートクラスのベースレートの合計に親レートクラスのベースレートを足したものの最大レートが決まります。

◆注

*子レートクラスは親レートクラスの上限を借用することができます。詳細については、**親クラスの指定**（16-8 ページ）を参照してください。*

バーストサイズの指定

レートクラスが制御するトラフィックフローのレートがベースレートを超過できるようにするには、[**Burst Size**] 設定を使用します。ベースレートの超過は、***バースト***と呼ばれます。レートクラスにバーストを許可するように設定すると（**0** 以外の値を指定）、BIG-IP は未使用の帯域幅を確保して、この帯域幅を後で使用して、トラフィックフローのレートが一時的にベースレートを超過できるようにします。バーストサイズの指定は、HTTP トラフィックのように、トラフィックパターンが変動したり、ベースレートを超過したりする場合の平準化に役立ちます。

[**Burst Size**] 設定の値は、バーストに許可される最大バイト数を定義します。したがって、バーストサイズを 5,000 バイトに設定すると、トラフィックフローのレートはベースレートを 1,000 バイト毎秒超過し、BIG-IP は最大 5 秒間、トラフィックをバーストできます。

バーストサイズを指定すると、BIG-IP はそのサイズのバーストの保管域を作成します。***バースト保管域***には、BIG-IP が後でバーストに使用する帯域幅が保存されます。バースト保管域は、トラフィックフローのレートがベースレートを超過すると減少し、トラフィックレートがベースレート以下になると補給されます。したがって、レートクラスで設定した [**Burst Size**] の値は、次のことを表します。

- トラフィックフローレートがベースレートを超過したときに、レートクラスが転送できる最大バイト数
- BIG-IP がバースト保管域に補給できる最大バイト数
- ベースレートを超えてバーストできる帯域幅の初期量

バーストサイズは、バイト単位で計測されます。たとえば、**10000** または **10K** は、10,000 バイトと等しくなります。デフォルト値は **0** です。

バースト保管域の消費

トラフィックフローのレートがベースレートを超過すると、BIG-IP は自動的にバースト保管域を消費します。このレートは、トラフィックフローがベースレートを超過するバイト数（毎秒）で決定されます。

トラフィックフローがベースレートを毎秒 1,000 バイト超過する以前の例で説明すると、トラフィックフローレートがベースレートを 2 秒間だけ超過した場合、2,000 バイトがバーストサイズから消費され、バーストできる最大バイト数は 3,000 に減少します。

バースト保管域の補給

トラフィックフローのレートがベースレート以下になると、BIG-IP は未使用の帯域幅（ベースレートと実際のトラフィックフローレートの差）をバースト保管域に確保します。後で、トラフィックフローがベースレートを超過したときに、BIG-IP はこの帯域幅を使用します。BIG-IP は、トラフィックフローがベースレートを超過したためにバースト保管域が消費された場合、保管域を補給します。

バースト保管域のサイズは、指定されたバーストサイズを超過できません。この理由により、BIG-IP は保管域が [**Burst Size**] 設定で指定された値に達するまで、未使用の帯域幅で保管域を補給します。

このため、バーストサイズが **5,000** に設定されている場合、BIG-IP はトラフィックフローのレートがベースレートを超過した場合に使用するための未使用の帯域幅を、5,000 バイトだけ確保できます。

◆注

バーストサイズを指定しても、レートクラスが上限を超えることは許されません。

0 以外のバーストサイズの指定

次の例では、**[Burst Size]** 設定を **0** 以外の値に設定した場合の BIG-IP の動作を説明します。

この例では、スループット の単位にデフォルト のビット 毎秒ではなく、バイト 毎秒を使用しています。これは単に例を簡単にするためです。ビット 毎秒を 8 で割ると、ビット 毎秒をバイト 毎秒に換算できます。

レートクラスを次のような値で設定したとします。

- ベースレート : 1,000 バイト毎秒
- 上限レート : 4,000 バイト毎秒
- バーストサイズ : 5,000 バイト

次のようなシナリオを考えます。

- **トラフィックフローが現在 800 バイト毎秒の場合**
  トラフィックフローのレートは、レートクラスで定義されたベースレート以下なので、バーストは必要ありません。

  トラフィックはベースレートより 200 バイト毎秒下回って流れているため、BIG-IP はバースト保管域に 200 バイトの未使用帯域幅を追加できます。ただし、まだバーストは発生していないため、保管域は指定された 5,000 バイトまでフルの状態であり、BIG-IP が未使用の帯域幅200 バイトを保管域に保存することはできません。この場合、BIG-IP は未使用の帯域幅を破棄します。

- **トラフィックが 1,000 バイト毎秒まで増加した場合（ベースレートと同等）**
  まだバーストは発生せず、未使用の帯域幅もありません。

- **トラフィックが 2,500 バイト毎秒まで上昇した場合**
  トラフィックが毎秒 2,500 バイトで流れ続けると 1 秒ごとに、BIG-IP はバースト保管域の 1,500 バイト（トラフィックフローレートとベースレートとの差）を消費します。このため、このレートでは3秒と少しの間だけバーストしてから、バースト保管域の5,000 バイトを使い切ることになります。保管域が消費されると、BIG-IP はトラフィックフローレートをベースレートの1,000バイト毎秒まで低下させ、バーストの発生を抑制します。

- **トラフィックが 800 バイト毎秒まで低下した場合**
  バーストは不要ですが、保管域が空になっているため、BIG-IP はバースト保管域に未使用の帯域幅 200 バイト毎秒を追加できます。トラフィックが 800 バイト毎秒で流れ続けると、バースト保管域は 25 秒後には 0 から 5,000 バイトへと完全に満たされます（毎秒 200 バイト）。トラフィックが流れなくなると、未使用の帯域幅が

1,000 バイト毎秒となり、BIG-IP は 1,000 バイト毎秒をバースト保管域に追加し、5 秒間で保管域は 0 から 5,000 バイトいっぱいになります。

帯域幅の借用

場合によっては、レートクラスが親クラスのバースト保管域から帯域幅を借用することがあります。詳細については、次の項の「*親クラスの指定*」を参照してください。

方向の指定

[**Direction**] 設定を使用して、レートクラスをクライアントまたはサーバトラフィックに適用できます。これにより、レートクラスをクライアントへのトラフィック、サーバへのトラフィック、またはクライアントとサーバ両方へのトラフィックに適用できます。指定できる値は、[**Any**]、[**Client**]、または [**Server**] です。デフォルト値は [**Any**] です。

方向の指定は、トラフィックの性質上、方向に偏りがある場合に役立ちます。たとえば、FTP サービスを外部クライアントに提供する場合、サイトからファイルをダウンロードするクライアントのスループットの制限より、ファイルを自分のサイトにアップロードするクライアントのスループットを制限するほうが重要な場合があります。この場合、FTP レートクラスの方向として [**Server**] を選択します。[**Server**] の値は、クライアントからサーバに流れるトラフィックに対するスループット制約にのみ適用されるからです。

親クラスの指定

レートクラスを作成する場合、[**Parent Class**] 設定を使用して、レートクラスの親クラスを指定できます。これによって、子レートクラスは親クラスの上限から未使用の帯域幅を借用できるようになります。子クラスは、親の上限から未使用の帯域幅を借用できますが、親クラスは子クラスから借用することはできません。また、同じ親クラスを持つ 2 つの子クラス間の借用や、関係のないレートクラス間の借用も不可能です。

親クラスは、[New Rate Class] 画面を表示して、[**Advanced**] を選択し、[**Parent Class**] 設定でレートクラス名を選択して指定します。

親クラス自身にも、循環依存関係を作成しない限り、親を設定できます。***循環依存関係***とは、レートクラスが直接的または間接的にそれ自身の子となっている関係です。

レートクラスに親クラスがある場合、子クラスは親クラスの上限から未使用の帯域幅を取得できます。このプロセスは、次のようにして実行されます。

- 子クラスに適用されるトラフィックフローのレートがベースレートを超過すると、子クラスは前述したようにバースト保管域を消費しはじめます。
- 保管域が空の場合（またはバーストサイズがレートクラスに定義されていない場合）、BIG-IP は親クラスの上限から未使用のベースレート帯域幅を取得し、子クラスに与えます。

- 親クラスからの未使用の帯域幅が消費されると、子クラスは親クラスの保管域を使用しはじめます。
- 親クラスの保管域が空の場合（または親クラスにバーストサイズが定義されていない場合）、親クラスに親クラスがある場合は、子クラスは親クラスの親から帯域幅を借用しようとします。
- このプロセスは、借用できる帯域幅がなくなるか、借用できる親がなくなるまで続きます。

借用によって子のバースト期間が延長できるだけで、子クラスの上限が引き上げられるわけではありません。

◆注

上記の説明では「借用」という表現を使いましたが、子クラスが借用する帯域幅は後で親クラスに返されません。また、子クラスの未使用の帯域幅が親クラスに返却されることもありません。

シェーピングポリシーの指定

破棄ポリシーとキュー方式のカスタマイズ値を含むシェーピングポリシーを指定します。デフォルト値は **None** です。

シェーピングポリシーを追加作成するには、**bigpipe** ユーティリティ、または Traffic Management シェル（**tmsh**）を使用します。

キュー方式の指定

[**Queue Method**] 設定では、BIG-IP がパケットをデキューする方法と順序を決定します。

レートクラスでは、以下の 2 つのキュー方式をサポートしています。

- **Stochastic Fair Queue**
  SFQ（Stochastic Fair Queue）方式は、トラフィックを多数のリストのセットにキューイングし、定期的に変化する接続情報のハッシュに基づいて特定のリストを選択する、キューイングです。これによって、同じ接続からのトラフィックは、常に同じリストにキューイングされます。SFQ では、Round Robin 方式でリストのセットからトラフィックをデキューします。1 つの高速接続によってキューが独占され低速接続が犠牲になるということがないため、全体的にデキューが公平に実行されます。
- **プライオリティ付き FIFO**
  PFIFO（プライオリティ付き FIFO）は、すべてのトラフィックを、トラフィックの Type of Service（ToS）フィールドに基づいて 5 つのリストのセットにキューイングする方式です。リストのうち 4 つは、4 つの使用できる ToS の値、[**Minimum delay**]、[**Maximum throughput**]、[**Maximum reliability**]、[**Minimum cost**] に対応します。5 番目のリストは、**ToS** 値のないトラフィックを表します。PFIPO 方式では、これらの 5 つのリストを処理して、**ToS** フィールドの意味を可能な限り維持しようとします。たとえば、[**ToS**]

フィールドが **Minimum cost** に設定されているパケットは、[**ToS**] フィールドが [**Minimum delay**] に設定されているパケットよりキューイング解除のプライオリティが低くなります。

破棄ポリシーの指定

破棄ポリシーは、キューのトラフィックが一杯になった場合、必要に応じて、処理パケットをいつどのように破棄するかを指定します。デフォルト値は **tail** です。

指定できる値は次のとおりです。

- **fred**
  フロー内のトラフィックタイプに従い、パケットを破棄します。
- **red**
  パケットをランダムに破棄します。
  *
- **tail**
  トラフィックストリームの末尾を破棄します。

破棄ポリシーを追加作成するには、**bigpipe** ユーティリティ、または Traffic Management シェル（**tmsh**）を使用します。

レートクラスの管理

レートクラスを作成すると、設定ユーティリティを使用して既存のレートクラスをリストしたり、レートクラスの設定を表示または変更したり、レートクラスを削除したりできるようになります。

既存のレートクラスをリストするには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Rate Shaping**] をクリックします。
   既存のレートクラスのリストとその設定値が表示されます。
2. レートクラスのリストが表示されます。

レートクラスを表示または変更するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Rate Shaping**] をクリックします。
   既存のレートクラスのリストが表示されます。
2. リストのレートクラス名をクリックします。
   レートクラスの設定が表示されます。
3. 設定値を、そのまま使用するか、または変更します。レートクラスの設定については、「*レートクラス設定の設定*」（16-4 ページ）を参照してください。
4. [**Update**] をクリックします。

レートクラスを削除するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**Rate Shaping**] をクリックします。
   既存のレートクラスのリストが表示されます。
2. リストでレートクラス名を指定して、名前の左側の [Select] ボックスをチェックします。
3. 画面下部の [**Delete**] をクリックします。
   削除を確認する画面が表示されます。
4. [**Delete**] をクリックします。
   これにより、レートクラスが削除されます。

17

iRule の記述

- iRule の概要
- iRule の作成
- iRule 評価の制御
- iRule コマンドの使用
- プロファイルの処理
- iRule でのセッションパーシステンスの有効化
- データグループの作成、管理、使用

iRule の概要

iRule は BIG-IP® ローカルトラフィック管理システムの強力で柔軟な機能で、ネットワークトラフィックの管理に使用できます。業界標準の Tools Command Language（Tcl）に基づいた構文を使用して、iRule™ 機能では、ヘッダデータに基づいてプールを選択したり、定義したコンテンツの種類を検索してトラフィックをダイレクトしたりできます。このようにして、iRule 機能によって、ニーズに最適な方法でコンテンツの切り替えをカスタマイズする能力が、大幅に向上できます。

この概要の後半では、iRule の概略を説明し、iRule を構成する基本要素の一覧を示し、iRule を使用してトラフィックを特定の宛先（プールや特定のノードなど）にダイレクトする例をいくつか示します。

◆ 重要

iRule 構文の詳細については、F5 Networks DevCentral の Web サイト ***http://devcentral.f5.com/Default.aspx?tabid=119*** *を参照してください。iRule は、標準 Tcl 文法規則に従う必要があります。Tcl 構文の詳細については、****http://tmml.sourceforge.net/doc/tcl/index.html*** *を参照してください。*

iRule について

iRule は、バーチャルサーバに定義したデフォルトのプール以外のプールに接続したい場合に記述するスクリプトです。iRule を使用することで、トラフィックをダイレクトする宛先をより直接的に指定できます。iRule を使用して、トラフィックをプールに送信するほか、個々のプールメンバ、ポート、URI にも送信できます。

作成した iRule は、コンテンツ切り替えの必要性に従って、簡素化または詳細化が可能です。図 17.1 に、簡単な iRule の例を示します。

```
when CLIENT_ACCEPTED {
  if { [IP::addr [IP::client_addr] equals 10.10.10.10] } {
     pool my_pool
 }
}
```

図 17.1 iRule の例

iRule は、クライアント側の接続が受理されるとトリガされ、クライアントのアドレスが **10.10.10.10** に一致すると、BIG-IP はパケットをプール **my_pool** に送信します。

UIE（Universal Inspecton Engine）と呼ばれる機能を使用して、パケットのヘッダまたは実際のパケットのコンテンツを検索して、その検索結果に従ってパケットをダイレクトする iRule を作成できます。iRule は、クライアント認証の試行結果に従って、パケットをダイレクトすることもできます。

iRule はトラフィックを特定のプールにダイレクトするほか、ポート番号やURIパスを含む個々のプールメンバにもダイレクトできます。これによって、パーシステンスを実装したり、特定のロードバランシング要求を満たすことができます。

iRule を記述する構文は、Tcl（Tool Command Language）プログラミング標準に従っています。このため、多数の標準的な Tcl コマンドを使用でき、BIG-IP がロードバランシングの効率をより高めることができるように、堅牢な拡張機能を使用することもできます。

基本 iRule 要素

iRule は、これらの基本要素から構成されます。

- イベント宣言
- 演算子
- iRule コマンド

イベント宣言

iRule はイベント駆動型で、BIG-IP は iRule で指定したイベントに基づいて iRule をトリガします。***イベント宣言***は、そのイベントが発生したときに BIG-IP が iRule をトリガする、iRule 内でのイベントの指定です。iRule をトリガするイベント宣言の例として **HTTP_REQUEST** があり、これによって、BIG-IP が HTTP リクエストを受信すると iRule がトリガされます。また、**CLIENT_ACCEPTED** では、クライアントが接続を確立すると、iRule がトリガされます。

図 17.2 は、iRule でのイベント宣言の例を示します。

```
when HTTP_REQUEST {
  if { [HTTP::uri] contains "aol" } {
     pool aol_pool
  } else {
     pool all_pool
 }
}
```

図 17.2 iRule 内でのイベント宣言の例

iRule イベントの詳細については、「*イベントの指定*」（17-9 ページ）を参照してください。

演算子

iRule 演算子は、式の中の 2 つのオペランドを比較します。Tcl 標準演算子を使用するほか、表 17.1 に挙げる演算子も使用できます。

演算子	構文
関係演算子	**contains** **matches** **equals** **starts_with** **ends_with** **matches_regex**
論理演算子	**not** **and** **or**

***表 17.1** iRule 演算子*

たとえば、**contains** 演算子を使用して、変数オペランドを定数と比較できます。これを実行するには、「HTTP URI に **aol** が含まれている場合、プール **aol_pool** に送信する」を表現する **if** ステートメントを作成します。図 17.2（17-2 ページ）は、この動作を実行する iRule の例を示します。

iRule コマンド

iRule 内の ***iRule コマンド***によって、BIG-IP はデータのクエリ、データの操作、トラフィックの宛先の指定などのアクションを実行します。次の種類のコマンドを iRule 内に指定できます。

- **ステートメントコマンド**
  これらのコマンドによって、トラフィックの宛先の選択や、SNAT 変換アドレスの割り当てなどのアクションが実行されます。ステートメントコマンドの例として **pool <name>** があり、これは指定したロードバランシングプールにトラフィックをダイレクトします。詳細については、「*iRule コマンドの使用*」（17-12 ページ）を参照してください。
- **データをクエリまたは操作するコマンド**
  一部のコマンドは、ヘッダとコンテンツデータを検索しますが、その他のコマンドは、HTTP リクエストへのヘッダの挿入など、データ操作を実行します。クエリコマンドの例には **IP::remote_addr** があり、これは接続のリモート IP アドレスを検索して返します。データ操作コマンドの例には、**HTTP::header remove <name>** があります。このコマンドは、指定したヘッダの最後に使用された箇所を要求または応答から削除します。
- **ユーティリティコマンド**
  これらのコマンドは、コンテンツの解析と操作に役立つ機能です。ユーティリティコマンドの例には **decode_uri <string>** があり、指定した文字列を **HTTP URI** エンコーディングを使用してデコードし、結果を返します ユーティリティコマンドの詳細については、「*ユーティリティコマンド*」（17-13 ページ）を参照してください。

トラフィックの宛先とアドレス変換の指定

前のセクションで説明したように、iRule コマンドは BIG-IP に対して直接様々な操作を指示できます。以降のセクションでは、トラフィックを特定の宛先にダイレクトしたり、SNAT 実装の変換アドレスを割り当てたりする iRule コマンドの例を示します。

ロードバランシングプールの選択

iRule でクエリを指定すると、**pool** コマンドを使用して、BIG-IP が要求を送信するロードバランシングプールを選択できます。図 17.3 は、このコマンドの例を示します。

```
when HTTP_REQUEST {
  set uri [HTTP::uri]
  if { $uri ends_with ".gif" } {
     pool my_pool
  } elseif { $uri ends_with ".jpg" } {
     pool your_pool
 }
}
```

***図 17.3** iRule 内での **pool** コマンドの例*

特定のサーバの指定

pool コマンドの代わりに、トラフィックを特定のサーバにダイレクトする iRule を記述できます。これには、**node** コマンドを使用します。図 17.4 は、このコマンドの例を示します。

```
when HTTP_REQUEST {
  if { [HTTP::uri] ends_with ".gif" } {
     node 10.1.2.200 80
 }
}
```

***図 17.4** iRule 内での **node** コマンドの例*

キャッシュサーバのプールの選択

キャッシュサーバのプールからサーバを選択する iRule を作成できます。図 17.5 は、キャッシュサーバのプールからサーバを選択する iRule を示します。

```
when HTTP_REQUEST
   # This line specifies the expressions that determine whether the BIG-IP system sends
requests to the cache pool:
   if { [HTTP::uri] ends_with "html" or [HTTP::uri] ends_with "gif" } {
      pool cache_pool
      set key [crc32 [concat [domain [HTTP::host] 2] [HTTP::uri]]]
      set cache_mbr [persist lookup hash $key node]
      if { $cache_mbr ne "" } {
        # This line verifies that the request is not coming from the cache:
        if { [IP::addr [IP::remote_addr] equals $cache_mbr] }
          # This line sends the request from the cache to the origin pool:
             pool origin_pool
             return
          }
       }
       # These lines ensure that the persistence record is added for this host/URI:
       persist hash $key
    } else {
       pool origin_pool
    }
}
```

***図 17.5** キャッシュプールのサーバへのロードバランシング*

BIG-IP は、プールメンバが利用不可になったときではなく、BIG-IP が URI の要求を受信したときに、URI を新しいキャッシュメンバにリダイレクトします。

HTTP リクエストのリダイレクト

iRule を特定のプールを選択するように設定するほか、**HTTP::redirect** iRule コマンドを使用して、HTTP リクエストを特定の場所にリダイレクトするように iRule を設定することもできます。場所には、ホスト名または URI を指定できます。

図 17.6 は、HTTP 応答をリダイレクトするように設定された iRule の例を示します。

```
when HTTP_RESPONSE {
  if { [HTTP::status] contains "404"} {
    HTTP::redirect "http://www.siterequest.com/"
 }
}
```

***図 17.6** HTTP リダイレクションに基づく iRule*

図17.7は、HTTPリクエストをリダイレクトするiRuleの例を示します。

```
when HTTP_REQUEST {
  if { [HTTP::uri] contains "secure"} {
    HTTP::redirect "https://[HTTP::host][HTTP::uri]"
 }
}
```

図 17.7 HTTP リダイレクションに基づく、別の iRule

SNAT 接続への変換アドレスの割り当て

iRule 機能には、**snat** と **snatpool** という、2 つのステートメントコマンドがあります。**snat** コマンドを使用して、設定ユーティリティの [SNAT] 画面を使用しなくても、iRule の中から、特定の変換アドレスを元の IP アドレスに割り当てることが可能です。

snatpool コマンドを使用しても、変換アドレスを元の IP アドレスに割り当てられますが、**snat** コマンドとは異なり、**snatpool** コマンドでは以前作成した特定の SNAT プールから BIG-IP が変換アドレスを選択します。

SNAT 実装の詳細については、第 14 章「*SNAT の設定*」を参照してください。

iRule の作成

iRule は、設定ユーティリティを使用して作成します。

◆ 重要

*iRule が作成されると、BIG-IP はその iRule を現在の管理パーティションに配置します。パーティションの詳細については、『**TMOS™ Management Guide for BIG-IP® Systems**』を参照してください。*

iRule を作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**iRules**] をクリックします。
   [iRules] 画面が開きます。
2. 右上の [**Create**] をクリックします。
   ***注**: **[Create]** ボタンをクリックできない場合は、現在のユーザロールでは iRule を作成する権限が与えられていないことを示します。*
3. [**Name**] ボックスに、1 ～ 31 文字の名前を入力します。
4. [**Definition**] ボックスに、iRule の構文を入力します。
5. [**Finished**] をクリックします。

iRule を作成する場合の構文の詳細については、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。

◆ 重要

*iRule を作成したら、iRule を参照するバーチャルサーバを設定する必要があります。iRule を参照するバーチャルサーバの設定の詳細については、第 2 章「**バーチャルサーバの設定**」を参照してください。*

iRules と管理パーティションについて

以下にあげる iRule の設定コンセプトは管理パーティションと関連があるため、しっかり理解しておく必要があります。

- iRule では、対象オブジェクトがどのパーティションに置かれているかに関係なく、あらゆるオブジェクトを参照できるようになっています。たとえば、**partition_a** に置かれた iRule に、**partition_b** にあるプールを指定するプールステートメントが含まれる場合もあります。
- iRule の割り当ては、現在の Write パーティションまたは **Common** パーティションに置かれたバーチャルサーバからのみ削除することができます。
- iRule を関連付けられるのは、現在の Write パーティションまたは **Common** パーティションに置かれたバーチャルサーバにかぎられます。

- 既存の 1 つの iRule を複数のバーチャルサーバと関連付けることが可能です。その場合の iRule は、現在の Write パーティション内の各バーチャルサーバに関連付けられた iRule ということになります。このコマンドを実行すると、それまでの iRule 割り当てがすべて上書きされるため、F5 ではこのコマンドの使用を推奨していません。

iRule 評価の制御

iRule を使用しない基本的なシステム設定では、BIG-IP は、受信トラフィックを、そのトラフィックを受信するバーチャルサーバに割り当てられたデフォルトのプールにダイレクトします。しかし、BIG-IP で、一部の接続を他の宛先にダイレクトさせたい場合があります。これを実行するには、特定の種類のイベントが発生した場合に、トラフィックを別の宛先にダイレクトする iRule を記述します。それ以外の場合は、トラフィックはバーチャルサーバに割り当てられたデフォルトプールに送信されます。

したがって iRule は、iRule で指定されたイベントが発生した場合にいつも評価されることになります。たとえば、iRule にイベント宣言 **CLIENT_ACCEPTED** が含まれている場合、iRule は、BIG-IP がクライアント接続を受け入れるとトリガされます。BIG-IP は、iRule の残りの指示に従い、パケットの宛先を決定します。

設定の前提条件

BIG-IP が、作成した iRule を評価する前に、次を実行する必要があります。

- **iRule をバーチャルサーバに割り当てる**
  iRule がバーチャルサーバに割り当てられると、バーチャルサーバがプールまたはプロファイルを参照するのと同様の方法で、バーチャルサーバが iRule を参照することになります。
- **バーチャルサーバが適切なプロファイルを参照するようにする**
  たとえば、iRule にイベント宣言 **HTTP_REQUEST** が含まれる場合、BIG-IP は、バーチャルサーバが **http** プロファイルタイプを参照する場合にのみ iRule を評価します。

◆注

*イベント **HTTP_REQUEST** を指定する iRule を割り当てる場合、バーチャルサーバが適切なプロファイルタイプを参照するようにします。*

iRule とプロファイルのバーチャルサーバへの割り当ての詳細については、第 2 章「*バーチャルサーバの設定*」を参照してください。

イベントの指定

iRule 機能には、iRule で使用できるいくつかの種類のイベント宣言があります。イベント宣言を指定すると、BIG-IP が iRule を評価するタイミングが決まります。以降のセクションでは、これらのイベントタイプの一覧と説明を示します。また、iRule コンテキストの概念と、**when** キーワードの使用方法についても説明します。

イベントタイプ

iRule コマンド構文には、iRule で指定できるいくつかのイベントタイプの宣言があります。たとえば、次のようになります。

- グローバルイベント（**CLIENT_ACCEPTED** など）
- HTTP イベント（**HTTP_REQUEST** など）
- SSL イベント（**CLIENTSSL_HANDSHAKE** など）
- 認証イベント（**AUTH_SUCCESS** など）

iRule イベントとその説明の詳細については、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。

iRule のコンテキスト

iRule に指定する各イベントに対して、キーワード **clientside** または **serverside** で示す、コンテキストを指定できます。各イベントには関連付けられたデフォルトコンテキストがあるため、コンテキストをデフォルトから変更するには、コンテキストを宣言するだけです。

たとえば、図 17.8 に示す **my_iRule1** では、イベント宣言 **CLIENT_ACCEPTED** と、iRule コマンド **IP::remote_addr** があります。この場合、iRule コマンドが返す IP アドレスはクライアントのものです。イベント宣言 **CLIENT_ACCEPTED** のデフォルトコンテキストが **clientside** だからです。

```
when CLIENT_ACCEPTED {
  if { [IP::addr [IP::remote_addr] equals 10.1.1.80] } {
     pool my_pool1
 }
}
```

***図 17.8** デフォルトコンテキスト **clientside** を使用する iRule*

同様に、イベント宣言 **SERVER_CONNECTED** と iRule コマンド **IP::remote_addr** を iRule に追加すると、iRule コマンドが返す IP アドレスはサーバのアドレスになります。イベント宣言 **SERVER_CONNECTED** のデフォルトコンテキストは **serverside** だからです。

図 17.8 は、iRule コマンドの処理にデフォルトコンテキストを使用する iRule を記述した場合の状態を示します。ただし、**clientside** および **serverside** キーワードを明示的に指定して、iRule コマンドの動作を変更することもできます。

前の例を使用すると、図 17.9（17-11 ページ）では、イベント宣言 **SERVER_CONNECTED** が示され、iRule コマンド **IP::remote_addr** のキーワード **clientside** が明示的に指定されています。この場合、イベント宣言のデフォルトコンテキストが serverside であっても、iRule コマンドが返す IP アドレスはクライアントのものです。

```
when SERVER_CONNECTED {
  if { [IP::addr [IP::addr [clientside {IP::remote_addr}] equals 10.1.1.80] } {
     discard
  }
 }
```

図 17.9 コンテキストを明示的に指定する iRule

when キーワードの使用

iRule でのイベント宣言は、**when** キーワードに続けてイベント名を使用して作成します。前の図で、iRule でのイベント宣言の例を示しています。

バーチャルサーバでの iRule のリスト

複数の iRule をバーチャルサーバにリソースとして割り当てる場合、バーチャルサーバに iRule をリストする順序を考慮することが重要です。これは、重複した iRule イベントがある場合、該当する iRule のリスト順に BIG-IP が処理を行うためです。iRule イベントは、イベントのトリガを終了でき、BIG-IP が後続のイベントをトリガできなくします。

◆注

iRule がプロファイルを参照する場合、BIG-IP は、バーチャルサーバに割り当てられた iRule のリストでの順序に関係なく、このタイプの iRule を最後に処理します。

iRule コマンドの使用

iRule コマンドには、以下の 3 つの種類があります。

- ステートメントコマンド
- クエリおよび操作コマンド
- ユーティリティコマンド（関数とも呼ばれます）

ステートメントコマンド

iRule で使用できるコマンドのなかには、ステートメントコマンドと呼ばれるものがあります。*ステートメントコマンド*によって、BIG-IP はさまざまなアクションを実行できます。たとえば、これらのコマンドには、BIG-IP がトラフィックをダイレクトするプールまたはサーバを指定できるものがあります。その他には、SNAT 接続を実装する変換アドレスを指定するコマンドがあります。また、データグループまたはパーシステンスプロファイルなどのオブジェクトを指定するものもあります。

ステートメントコマンドの詳細については、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。

クエリおよび操作コマンド

iRule コマンドを使用すると、要求または応答のヘッダやコンテンツに含まれる特定のデータを問い合わせたり、そのデータを操作することができます。データ操作とは、データの挿入、置換、削除、およびヘッダおよび Cookie で見つかった値の設定のことです。

たとえば、iRule 内で **IP::idle_timeout** コマンドを使用して、パケットヘッダに設定された現在のアイドルタイムアウトを問い合わせ、そのパケットでロードバランシングを行うことができます。また、**IP::idle_timeout** コマンドを使用すると、選択した値にアイドルタイムアウトを設定することができます。

iRule クエリおよび操作コマンドは、***namespaces（ネームスペース）***と呼ばれるカテゴリに分類されています。グローバルネームスペースのコマンドを除き、各 iRule クエリまたは操作コマンドは、コマンド名にネームスペースを含んでいます。たとえば、**IP** ネームスペースのコマンドのひとつに **IP::idle_timeout** があります。また、**HTTP** ネームスペースのコマンドの例として、**HTTP: header** があります。

iRule コマンドのネームスペースの一覧については、F5 Networks DevCentral の Web サイト **http://devcentral.f5.com/Default.aspx?tabid=119** を参照してください。

ユーティリティコマンド

BIG-IP には、iRule 内で使用できる多数のユーティリティコマンドがあります。これらのコマンドを使用して、コンテンツの解析や取得、データの ASCII フォーマットへのエンコード、データ整合性の検証、アクティブプールおよびプールメンバに関する情報の取得などを実行できます。

プロファイルの処理

iRuleを記述する場合、iRuleに特定のプロファイル設定の値を通知し、その情報に適したトラフィック管理を実行できるようにしたいことがあります。iRule 機能には、iRule 内で指定したプロファイル設定の値を読み込む、専用のコマンドがあります。

iRule ではプロファイル設定の値を読み込むほか、特定の設定の値をオーバーライドできます。これにより、バーチャルサーバ経由で確立した接続に対し BIG-IP が適用する値とは別の値を、個別の接続に適用するように設定できます。

プロファイル設定の読み込み

iRule 機能には、**PROFILE** というコマンドがあります。**PROFILE** コマンドを iRule で指定してプロファイルタイプと設定を指定すると、iRule はそのプロファイル設定の値を読み込みます。このために iRule は、バーチャルサーバに割り当てられた、指定されたプロファイルタイプを検索し、**PROFILE** コマンドシーケンスで指定された設定の値を読み込みます。iRule は、この情報を使用してトラフィックを管理します。

たとえば、iRule で **PROFILE::tcp idle_timeout** コマンドを指定できます。BIG-IP は、バーチャルサーバ（**my_tcp** など）に割り当てられた TCP プロファイルを見つけ、[**Idle Timeout**] 設定に割り当てた値に対してクエリを実行します。

プロファイル設定のオーバーライド

ヘッダおよびコンテンツデータのクエリおよび操作のための iRule コマンドには、様々なプロファイルで同じ設定を持つものがあります。これらのコマンドを iRule で使用し、イベントが iRule をトリガすると、BIG-IP は iRule で指定された値を使用してこれらのプロファイル設定の値をオーバーライドします。

たとえば、HTTP プロファイルでは HTTP データの圧縮に使用するバッファサイズを指定しますが、特定のタイプの HTTP 接続に異なるバッファサイズを指定したい場合があります。この場合、iRule でコマンド **HTTP::compress_buffer_size** を使用し、プロファイルの値とは別の値を指定します。

iRule でのセッションパーシステンスの有効化

第 7 章「*セッションパーシステンスの有効化*」では、パーシステンスプロファイルを設定してバーチャルサーバに割り当てて、セッションパーシステンスを有効にする方法を説明します。この章で説明するように、BIG-IP はこれらのパーシステンスプロファイル設定を、バーチャルサーバを経由する該当セッションすべてに適用します。たとえば、**msrdp** プロファイルをバーチャルサーバに割り当てると、BIG-IP はこの設定を受信するすべての Microsoft® Remote Desktop Protocol（RDP）接続に適用します。

ただし、よりきめ細かなレベルでパーシステンスを有効にしたい場合もあります。たとえば、終了していない SSL トラフィックのみで動作する **ssl** パーシステンスプロファイルを使用する代わりに、HTTP リクエスト ヘッダに挿入した SSL 証明書の状態に基づいてセッションをパーシステンスにしたいことがあります。このためには、**HTTP::header** コマンドを使用して iRule を記述し、iRule をバーチャルサーバに割り当てます。BIG-IP が SSL 要求を受け取ると、iRule は要求にヘッダとして証明書の状態を挿入し、その状態に基づいてセッションをパーシステンスします。

BIG-IP には、特殊な iRule コマンド **persist** があります。このコマンドは、第 7 章「*セッションパーシステンスの有効化*」に記載したタイプのセッションパーシステンスを実装します。使用方法は、iRule に **persist** コマンドを入力して、パーシステンスタイプを指定するだけです。

たとえば、特定のイベントが発生したときに SSL 接続をパーシステンスする iRule を、セッション ID のパーシステンスを基にして記述できます。図 17.10 は、これを実現する iRule の記述例を示します。

```
when CLIENTSSL_HANDSHAKE {
    persist ssl
}
```

***図 17.10** セッション ID に基づく SSL パーシステンスを実現する iRule の例*

persist none、**hash**、**srcaddr**、**destaddr**、および **uie** コマンドはどのような場合でも使用でき、パーシステンスプロファイルが設定されておらず、バーチャルサーバに割り当てられていない場合でも使用できます。ただし、**persist ssl**、**cookie**、**msrdp**、および **sip** コマンドでは、対応するパーシステンスプロファイルをバーチャルサーバに割り当てる必要があります。これらのコマンドを、対応するプロファイルがない場合に使用すると、iRule のランタイムエラーが発生します。パーシステンスプロファイルのバーチャルサーバへの割り当ての詳細については、第 2 章「*バーチャルサーバの設定*」を参照してください。

データグループの作成、管理、使用

データグループは、iRule の記述に役立ちます。***データグループ***は、AOL クライアントの IP アドレスのセットなど、関連する要素のグループに過ぎません。データグループを **matchclass** コマンドまたは **contains** 演算子で指定すると、iRule 式内で複数の値を引数として並べる必要がなくなります。

BIG-IP では、**private_net**、**images**、**aol** という 3 つのデータグループが事前設定されています。

データグループの有用性を理解するには、まず、**matchclass** コマンドと **contains** 演算子について理解すると簡単です。

◆注

ユーザロールとパーティションアクセス割り当てに基づき、管理権限があるデータグループのみを管理することができます。

◆警告

*3 つの設定済データグループ（**private_net**、**images**、**aol**）の変更や削除は行えません。もし行うと悪影響が生じる可能性があります。*

matchclass コマンドの使用

BIG-IP には **matchclass** という iRule コマンドがあり、これを使用すると、iRule で使用されるコマンドが特定のデータグループのメンバを表現しているかどうかによって、プールを選択できます。**matchclass** コマンドを使用すると、BIG-IP は識別子に続く文字列がデータグループの名前であると解釈します。

たとえば、**matchclass** コマンドを使用して、**IP::remote_addr** コマンドの値がデータグループ AOL のメンバである場合に、BIG-IP で受信する AOL 接続をプール **aol_pool** にロードバランスすることができます。図 17.11 は、このタイプの iRule の例を示します。この場合、**matchclass** コマンドは、**aol** というオブジェクトが値のコレクション（つまり、データグループ）であることを示します。

```
when CLIENT_ACCEPTED {
  if { [matchclass [IP::remote_addr] equals $::aol] } {
     pool aol_pool
  } else {
     pool all_pool
 }
}
```

図 17.11 matchclass コマンドに基づく iRule

[IP::remote_addr] equals matchclass $::aol のような式は、式がデータグループの特定の値を少なくとも 1 つ持つ場合に True になります。

データグループの作成

matchclass コマンドを iRule で使用する場合、3 つのタイプのデータグループを指定できます。

- アドレスデータグループ ー IP アドレスのコレクション
- 文字列データグループ ー ***.jpg** など、文字列のコレクション
- 整数データグループ ー 数値のコレクション

以降の項では、これらのデータグループについて説明します。

◆注

データグループのサイズは、システムリソースによってのみ制限されます。

アドレスデータグループ

IP アドレスデータグループには、ネットワーク IP アドレスとホスト IP アドレスの、2 つのタイプがあります。

次の手順では、ネットワークまたはホストアドレスデータグループを作成します。

アドレスデータグループを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**iRules**] をクリックします。
   [iRules] 画面が開きます。
2. メニューバーで、[**Data Group List**] をクリックします。
3. 画面の右上で [**Create**] をクリックします。
4. [**Name**] ボックスで、**my_address_group** などの、データグループの一意の名前を入力します。
5. [**Type**] ボックスで、[**Address**] を選択します。
   画面が展開され、詳細な設定が表示されます。
6. [Records] セクションで、[**Host**] または [**Network**] を選択します。
7. [**Address**] ボックスに、データグループの最初の IP アドレスを入力します。ネットワークデータグループを作成する場合、[**Mask**] ボックスにマスクも入力します。
8. [**Add**] をクリックします。
   入力内容が [**Address Records**] ボックスに表示されます。
9. ステップ 7 とステップ 8 を繰り返して、すべての IP アドレスを入力します。
10. [**Finished**] をクリックします。

文字列データグループ

文字列データグループには、***.jpg** や ***.gif** などの、文字列のリストが含まれます。次の手順では、文字列データグループを作成します。

文字列データグループを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**iRules**] をクリックします。
   [iRules] 画面が開きます。
2. メニューバーで、[**Data Group List**] をクリックします。
3. 画面の右上で [**Create**] をクリックします。
4. [**Name**] ボックスで、**my_images** などの、データグループの一意の名前を入力します。
5. [**Type**] ボックスで、[**String**] を選択します。
   画面が展開され、文字列固有の設定が表示されます。
6. [**String**] ボックスに、データグループの最初の文字列を入力します。
7. [**Add**] をクリックします。
   入力内容が [**String Records**] ボックスに表示されます。
8. ステップ 6 とステップ 7 を繰り返して、すべての文字列を入力します。
9. [**Finished**] をクリックします。

整数データグループ

整数データグループには、整数のリストが含まれます。次の手順では、整数データグループを作成する方法を示します。

整数データグループを作成するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**iRules**] をクリックします。
   [iRules] 画面が開きます。
2. メニューバーで、[**Data Group List**] をクリックします。
3. 画面の右上で [**Create**] をクリックします。
4. [**Name**] ボックスで、**my_integer_group** などの、データグループの一意の名前を入力します。
5. [**Type**] ボックスで、[**Integer**] を選択します。
   画面が展開され、[Records] セクションが表示されます。
6. [**Integer**] ボックスに、データグループの最初の整数を入力します。
7. [**Add**] をクリックします。
   入力内容が [**Integer Records**] ボックスに表示されます。
8. ステップ 6 とステップ 7 を繰り返して、すべての整数を追加します。
9. [**Finished**] をクリックします。

ストレージオプション

BIG-IP では、インラインまたは外部の 2 つの方法で、データグループを保存できます。

インラインストレージ

データグループを作成した場合、BIG-IP はその全体を **bigip.conf** ファイルに自動的に保存します。このタイプのストレージは、***インラインストレージ***と呼ばれます。

データグループ内のデータを更新する必要がある場合、データグループ全体を再ロードする必要があります。一般には、インラインストレージでは大規模データグループの検索に多大な負荷がかかるため、システムリソースを余分に使用します。また、インラインストレージではインクリメンタルにデータを更新する場合、データグループ全体を再ロードする必要があります。このため、BIG-IP では、外部に、すなわち **bigip.conf** ファイル以外に、データグループを保存する機能を搭載しています。

外部ストレージ

BIG-IP では、**bigip.conf** ファイル以外の別の場所にデータグループを保存するオプションがあります。このようなデータグループは、***外部データグループ***と呼ばれます。データグループは別の場所に外部的に保存されるため、**bigip.conf** ファイル自身にはデータグループのメタデータのみが含まれます。外部に保存されたデータグループファイルのデータは、値をカンマで区切りのリストで保存されます（CSV フォーマット）。

◆ 重要

*外部データグループのメタデータが含まれる **bigip.conf** ファイルをロードする場合、そのファイルが BIG-IP 9.4 より前のバージョンで作成されたものであれば、エラーが発生します。これは、外部データグループのメタデータに **extern** というキーワードが含まれ、これによってロード実行時にエラーが生じるためです。バージョン 9.4 以降の BIG-IP では、**bigip.conf** ファイルに **extern** キーワードが含まれなくてもいいようになっています。*

外部データグループの作成は、データがロードされるときにソートされないため、有効です。データは、カーネルのハッシュテーブルに保存されます。このストレージ方法により、iRule が大規模なデータグループを使用してトラフィックをプールにダイレクトするときのパフォーマンスが向上します。

外部データグループは、**/config** か **/var/class** のいずれかのディレクトリに配置する必要があります。外部データグループを保存するデフォルトの場所は、**/config** ディレクトリです。

外部データグループを実装するには、まずは図 17.12（17-20 ページ）で示すフォーマットに従い、テキストエディタでテキストファイルを作成します。次に、設定ユーティリティを使って、先に作成したテキストファイルを参照する **(External File)** タイプのデータグループを作成します。

データグループを外部に保存するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**iRules**] をクリックします。
   [iRules] 画面が開きます。
2. メニューバーで、[**Data Group List**] をクリックします。
3. 画面の右上で [**Create**] をクリックします。
4. [**Name**] ボックスで、外部の場所に保存したい既存のデータグループの名前を入力します。
5. [**Type**] ボックスで、[**(External File)**] を選択します。
6. [**Path/Filename**] ボックスに、外部ロケーションのパスとファイル名を入力します（例：**/var/class/my_address_group**）。
   このファイル名は、データグループ自身に割り当てた名前と一致する必要があります。

   ***注 : /config** ディレクトリ（デフォルトロケーション）にデータグループを保存する場合は、パス名で **/config** を指定する必要はなく、データグループのファイル名を入力するだけです。*
7. [**File Contents**] ボックスで、データグループに割り当てるファイルタイプを、[**Address**]、[**String**]、[**Integer**] から選択します。
8. [**Finished**] をクリックします。

BIG-IP は、外部データグループファイルのデータをカンマ区切りのリストとして保存し、IP アドレスなどのデータ値のフォーマットは、**bigip.conf** ファイルで使用されるフォーマットと同じになります。図 17.12 に、データグループファイル **/config/ip2.data group** の内容を示します。

```
network 195.93.32.0 mask 255.255.255.0,
network 195.93.33.0 mask 255.255.255.0,
network 195.93.34.0 mask 255.255.255.0,
network 195.93.48.0 mask 255.255.255.0,
network 195.93.49.0 mask 255.255.255.0,
network 195.93.50.0 mask 255.255.255.0
```

***図 17.12** 外部データグループファイルの例*

データグループプロパティの表示

設定ユーティリティを使用して、既存のデータグループのプロパティを表示できます。

データグループのプロパティを表示するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**iRules**] をクリックします。
   [iRules] 画面が開きます。
2. メニューバーで、[**Data Group List**] をクリックします。
3. データグループの名前をクリックします。
   そのデータグループのプロパティが表示されます。

データグループ メンバの管理

データグループがインラインで保存されている場合は、設定ユーティリティを利用して、メンバを既存のデータグループに追加したり、グループから削除することができます。

データグループが外部に保存されている場合は、**bigpipe** ユーティリティを使ってメンバの管理を行います。詳細については、『***Bigpipe Utility Reference Guide***』を参照してください。

データグループにメンバを追加するには

1. ナビゲーションペインの [Main] タブで、[**Local Traffic**] を展開して [**iRules**] をクリックします。
   [iRules] 画面が開きます。
2. メニューバーで、[**Data Group List**] をクリックします。
3. データグループの名前をクリックします。
   そのデータグループのプロパティが表示されます。
4. [Records] 領域で、該当するボックスにアドレス、文字列、整数を入力します。
5. [**Add**] をクリックします。
6. [**Update**] をクリックします。

A

ヘルスモニタとパフォーマンスモニタ

- モニタリファレンス

モニタリファレンス

ここから先は、ローカルトラフィック管理のヘルスモニタとパフォーマンスモニタについて説明していきます。モニタの説明はアルファベット順に記載しています。

モニタのコンセプトや手順に関する一般情報は、第 12 章「*モニタの設定*」を参照してください。

◆ 重要

*カスタムモニタの値を定義する場合は、予約済みキーワードの一覧にある値の使用は避けてください。詳しい情報は「Ask F5℠ Knowledge Base」（**https://support.f5.com**）のソリューション No 3653 をご覧ください（バージョン 9.0 以上）。*

External

モニタの External タイプを使用すると、独自のモニタタイプを作成できます。これには、External タイプのカスタムモニタを作成し、この中で実行するユーザ作成のモニタを指定します。

図 A.1 は、External タイプのモニタの設定とデフォルト値を示します。

```
Name ""
Type External
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
External Program ""
Arguments ""
Variables ""
Alias Address * All Addresses
Alias Service Port * All Ports
```

図 A.1 External タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_external_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **External Program**
  ユーザが提供するモニタプログラムの名前を指定する際に使用するのが、[**External Program**] 設定です。External タイプのモニタは、**/usr/bin/monitors** ディレクトリを検索します。

- **Arguments**
  [**Arguments**] 設定を使用して、必要なコマンドライン引数を指定することができます。
- **Variables**
  外部モニタで必要とされる変数「Name/Value ペア」を指定します。
- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

FirePass

FirePass モニタは、FirePass システムの健全性をチェックするためのものです。

図 A.2 に、FirePass タイプのモニタの設定とデフォルト値を示しています。

```
Name ""
Type External
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Cipher List HIGH:!ADH
Max Load Average 12.0
Concurrency Limit 95
User Name ""
Password ""
Alias Address * All Addresses
Alias Service Port * All Ports
```

図 A.2 FirePass タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_firepass_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。また、ノードの応答に RESET パケットが含まれる場合は、タイムアウトインターバルの設定値に到達する前に、そのノードに対してただちに **down** のフラグが立てられます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定するようにしてください。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Cipher List**
  暗号リストを指定しない場合、モニタはデフォルトの暗号リストである **DEFAULT:+SHA:+3DES:+kEDH** を使用します。
- **Max Load Average**
  モニタが FirePass システムの up/down のマーク付けに使用する数値を指定します。この設定値が FirePass の 1 分間の平均システム負荷と比較され、FirePass の平均システム負荷が [**Max Load Average**] 値を下回っていれば、モニタはその FirePass システムを up とマーク付けします。平均負荷が設定値より上の場合は、モニタはそのシステムを down とマーク付けします。デフォルト値は **12.0** です。
- **Concurrency Limit**
  ライセンスで許可された接続数の内、FirePass システムを up とマーク付けする最大値を指定します。たとえば、設定値が 95% の場合、モニタは許可された接続の 95% が使用中になるまで、FirePass システムを up とマーク付けします。許可された接続の 95% 以上が使用中になると、モニタはその FirePass システムを down とマーク付けします。デフォルト値は **95** です。
- **User Name** と **Password**
  パスワードによるセキュリティがない場合は、[**Username**] 設定と [**Password**] 設定に空白文字列 **""** を使用します。
- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

FTP

モニタの FTP タイプを使用すると、File Transfer Protocol（FTP）トラフィックを監視できます。このタイプのモニタは、指定されたファイルを **/var/temp** ディレクトリにダウンロードします。ファイルが取得されると、このチェックは成功です。ファイルが正常にダウンロードされた場合でも、BIG-IP には保存されません。

FTP モニタでは、ダウンロードされるファイルの**ユーザ名**、**パスワード**、およびファイルへのフルパスが指定されます。

図 A.3 は、FTP タイプのモニタの設定とデフォルト値を示します。

```
Name ""
Type FTP
Interval 10
Timeout 31
Manual Resume No
Check Until Up No
User Name ""
Password ""
Path/Filename ""
Mode Passive
Alias Address * All Addresses
Alias Service Port * All Ports
```

***図 A.3** FTP タイプのカスタムモニタとデフォルト値*

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_ftp_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 10 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 31 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

Gateway ICMP

Gateway ICMP タイプのモニタには、特殊な目的があります。このモニタは gateway failsafe を実装するプールに使用してください。

Gateway ICMP モニタは、プールメンバに適用できることを除いては、ICMP モニタと同じように機能します（ICMP モニタはノードのみに適用でき、プールメンバには適用できません）。図 A.4 は、設定済み **gateway_icmp** モニタの設定と値を示します。

```
Name ""
Type Gateway ICMP
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Transparent No
Alias Address * All Addresses
Alias Service Port * All Ports
```

図 A.4 gateway_icmp 設定済みモニタ

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_gw_icmp_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Transparent**
  Transparent モードはこのタイプのモニタのオプションであり、このモードを [**Yes**] に設定すると、モニタはモニタが関連付けられているノードに ping コマンドを発行します。**Transparent** モードの詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

HTTP

HTTP タイプのモニタを使用すると、Hypertext Transfer Protocol（HTTP）トラフィックの状態をチェックできます。TCP モニタと同様に、HTTP モニタは Web ページから特定のコンテンツを受信しようとしますが、TCP モニタとは異なり、ユーザ名とパスワードを送信することができます。図 A.5 は、設定済みモニタ **http** の設定を示します。

```
Name ""
Type HTTP
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Send String GET /
Receive String ""
User Name ""
Password ""
Reverse No
Transparent No
Alias Address * All Addresses
Alias Service Port * All Ports
```

図 A.5 http 設定済みモニタ

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_ftp_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Send String** と **Receive String**
  このタイプのモニタでは、[**Send String**] 値と [**Receive String**] 値が使われます。[**Send String**] 値が空白で、接続が確立できる場合、

サービスは **up** であるとみなされます。空白の [**Receive String**] 値はどのような応答とも一致します。コンテンツが [**ReceiveString**] 値と一致すればチェックは成功です。応答ヘッダの値は、[**Receive String**] 値として指定できます。たとえば、[**Receive String**] 属性の値は **404 Object Not Found** のようになります。

- **User Name** と **Password**
  パスワードによるセキュリティがない場合は、[**Username**] 設定と [**Password**] 設定に空白文字列 **""** を使用します。
- **Transparent** と **Reverse**
  Transparent モードと **Reverse** モードはいずれもオプションです。**Transparent** モードと **Reverse** モードの詳細については、第 12 章「*モニタの設定*」を参照してください。

HTTPS

Hypertext Transfer Protocol（HTTP）トラフィックの状態をチェックするには、HTTP タイプのモニタを使用します。HTTPS タイプのモニタは SSL セキュリティによって保護された Web ページから特定のコンテンツを受信しようとします。コンテンツが **Receive String 値**と一致すればチェックは成功です。

BIG-IP には、**https** と **https_443** という 2 つの設定済み HTTPS モニタが用意されています。図 A.6 は、設定済みモニタ **https** の設定を示し、図 A.7 は、設定済みモニタ **https_443** の設定を示します。

```
Name ""
Type HTTPS
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Send String GET /
Receive String ""
Cipher List ""
User Name ""
Password ""
Compatibility Enabled
Client Certificate ""
Reverse No
Transparent No
Alias Address * All Addresses
Alias Service Port * All Ports
```

図 A.6 https 設定済みモニタ

```
Name ""
Type HTTPS_443
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Send String GET /
Receive String ""
Cipher List ""
User Name ""
Password ""
Compatibility Enabled
Client Certificate ""
Reverse No
Transparent No
Alias Address * All Addresses
Alias Service Port HTTPS
```

図 A.7 https_443 設定済みモニタ

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_https_monitor**」や「**my_https_443_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Send String** と **Receive String**
  このタイプのモニタでは、[**Send String**] 値と [**Receive String**] 値が使われます。[**Send String**] 値が空白で、接続が確立できる場合、サービスは **up** であるとみなされます。空白の [**Receive String**] 値はどのような応答とも一致します。コンテンツが [**ReceiveString**] 値と一致すればチェックは成功です。応答ヘッダの値は、[**Receive String**] 値として指定できます。たとえば、[**Receive String**] 属性の値は **404 Object Not Found** のようになります。
- **Cipher List**
  暗号リストを指定しない場合、モニタはデフォルトの暗号リストである **DEFAULT:+SHA:+3DES:+kEDH** を使用します。
- **User Name** と **Password**
  パスワードによるセキュリティがない場合は、[**Username**] 設定と [**Password**] 設定に空白文字列 **""** を使用します。
- **Compatibility**
  [**Compatibility**] 設定を **Enabled** に設定すると、SSL オプションは ALL に設定されます。
- **クライアント証明書**
  [**Client Certificate**] 設定を使って、モニタがサーバに提示する証明ファイルを指定できます。

- **Transparent** と **Reverse**
  Transparent モードと **Reverse** モードはいずれもオプションです。**Transparent** モードと **Reverse** モードの詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

ICMP

モニタの ICMP タイプを使用すると、Internet Control Message Protocol（ICMP）を使用して簡単なノードチェックを実行できます。このチェックは、モニタが ICMP_ECHO ダイアグラムへの応答を受信すると成功です。図 A.8 は、設定済みモニタ **icmp** の設定項目と値を示します。

```
Name ""
Type ICMP
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Transparent No
Alias Address * All Addresses
```

図 A.8 icmp 設定済みモニタ

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_icmp_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Transparent**
  Transparent モードはこのタイプのモニタのオプションであり、このモードを **[Yes]** に設定すると、モニタはモニタが関連付けられているノードに ping コマンドを発行します。**Transparent** モードの詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Alias Address**
  [Alias Address] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

IMAP

IMAP タイプのモニタを使用すると、Internet Message Access Protocol（IMAP）トラフィックの状態をチェックできます。IMAP モニタは **Folder** 設定が追加された POP3 タイプのモニタです。モニタがサーバにログインでき、指定されたメールフォルダを開くことができればこのチェックは成功です。

図 A.9 は、IMAP タイプのモニタの設定とデフォルト値を示します。

```
Name ""
Type IMAP
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
User Name ""
Password ""
Folder INBOX
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図 A.9 IMAP タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_imap_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **User Name** と **Password**
  IMAP モニタでは、**ユーザ名**と**パスワード**を指定する必要があります。

- **Folder**
  [**Folder**] 設定では、モニタがサーバにログインした時にオープンするメールフォルダを指定します。モニタがサーバにログインでき、指定されたメールフォルダを開くことができればこのチェックは成功です。

- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。**Yes** と設定すると、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

◆注

通常、IMAP モニタによってチェックされるサーバでは、セキュリティを維持しながらモニタの認証も可能にするために特別な設定が必要です。

Inband

Inband タイプのモニタを使用する場合、BIG-IP はクライアントリクエストの一環としてパッシブモニタリングを実行します。Inband モニタは Standard タイプもしくは Performance タイプ（レイヤ 4）のバーチャルサーバと連動し、iRule の記述なしにパッシブモニタリングを実行することが可能です。

Inband タイプのモニタを設定すると、そのモニタがクライアントであれば、プールメンバに接続しようと試み、以下にように動作します。

- ユーザ指定の時間内にユーザが指定した回数だけ接続を試みたにもかかわらず、プールメンバが接続要求に応えない場合、モニタはそのプールメンバを **down** とマーク付けします。
- プールメンバが **down** とマーク付けされ、ユーザが指定した時間が経過すると、モニタは再度そのプールメンバへの接続を試みます（設定されている場合）。

F5 製品の強みは、Inband タイプのモニタをアクティブなモニタと組み合わせて使うことができる点です。

このタイプのモニタはシステムに負荷がほとんどかかりません。

◆注

*パッシブモニタリングの詳細については、第 12 章「**モニタの設定**」を参照してください。*

図 A.10 は、Inband タイプのモニタの設定とデフォルト値を示しています。

```
Name ""
Type Inband
Failures 3
Failure Interval 30
Response Time 10
Retry Time 300
```

***図 A.10** Inband タイプのカスタムモニタとデフォルト値*

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_inband_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Failures**
  [**Failures**] 設定では、プールメンバが down とマーク付けされるまで、指定された Failure Interval の間に送信される応答の失敗数を指定します。失敗の合計数は、接続の失敗の数、および指定された [**Response Time**] 値のインターバルの間にデータを返すことができなかった数を足したものとなります。デフォルト値は **3** です。なお、複数の TMM プロセスが実行される場合は、指定されたロードバランシング方式に応じて、プロセスごとの失敗数を使って計算を行います。たとえば、Round Robin ロードバランシング方式の場合、TMM プロセス数が **N**、プールメンバ数が **M** で、[**Failures**] 値

を **L** に設定したとすると、そのノードでは **down** とマーク付けされるまでに最大で **N*M*L+1** の失敗が発生してもいいことになります。

- **Failure Interval**
  [**Failure Interval**] 値はインターバルを秒単位で指定するものであり、このインターバル内で [**Failures**] 値で指定された数の失敗が発生すると、そのプールメンバは無効であるとマーク付けされます。
- **Response Time**
  [**Response Time**] 値は時間を秒単位で指定するものであり、この時間が過ぎてもプールメンバの応答がない場合、そのインターバルの失敗数は 1 つ増えることになります。値を **0** にすると、この設定は無効になります。
- **Retry Time**
  [**Retry Time**] 設定では、プールメンバが無効であるとマーク付けされた後に、システムがそのメンバに再度接続を試みるまでの時間を秒単位で指定します。接続に成功すると、そのプールメンバは **up** とマーク付けされます。値を **0** にすると、この設定は無効になります。

バーチャルサーバに関する注意事項

Inband モニタの実装に際しては、次のことに注意するようにします。

- Inband　モニタに関連付けることができるバーチャルサーバは、**Standard** と **Performance（レイヤ 4）**の 2 種類に限られる。
- バーチャルサーバの [**Protocol**] は **TCP** に設定すること。
- バーチャルサーバの [**HTTP Profile**] 設定は、**http** か **None**、もしくはカスタム HTTP プロファイルの名前に設定すること。

表 A.1（A-20 ページ）では、3 種類のバーチャルサーバ環境に Inband モニタを実装した場合のシステム動作の違いを示しています。

バーチャルサーバタイプ	動作の違い
Performance（レイヤ 4）	Inband モニタを設定し、Performance（レイヤ 4）タイプのバーチャルサーバとそのモニタを関連付けた環境では、以下の場合に **down** とマーク付けされます。 • 接続が確立できない。 • **tmm** インスタンスがサーバから RESET を受信した。 • クライアントがサーバの応答を受信しないまま、Inband モニタの [**Response Time**] 設定で指定した時間が過ぎてしまった（設定値が 0 でない場合）。 Inband モニタによってプールメンバが **down** とマーク付けされてから、[**Retry Time**] 設定で指定した秒数が経過すると（設定値が 0 でない場合）、そのプールメンバは要求を受信できるようになります。クライアントがプールメンバへの接続を問題なく確立・完了できる場合、そのプールメンバは **up** とマーク付けされます。

***表 A.1** バーチャルサーバタイプごとの Inband モニタの動作の違い*

バーチャルサーバタイプ	動作の違い
Standard （HTTP プロファイルなし）	Inband モニタを設定し、HTTP プロファイルを参照しない Standard タイプのバーチャルサーバとそのモニタを関連付けた環境では、プールメンバとの接続が確立されたか、プールメンバが要求に応答している場合（あるいはそれらの両方を満たす場合）、および TCP プロファイルで指定した最大再送数を超えた場合に **down** とマーク付けされます。 Inbandモニタによってプールメンバが**down**とマーク付けされてから、**[Retry Time]** 設定で指定した秒数が経過すると（設定値が 0 でない場合）、そのプールメンバは要求を受信できるようになります。クライアントがプールメンバとの接続を問題なく確立・完了できる場合、接続が完了した時点でそのプールメンバは **up** とマーク付けされます。
Standard （HTTP プロファイルあり）	Inband モニタを設定し、HTTP プロファイルを参照する Standard タイプのバーチャルサーバとそのモニタを関連付けた環境では、以下の場合に **down** とマーク付けされます。 • プールメンバとの接続が確立されたか、プールメンバが要求に応答している場合（あるいはそれらの両方を満たす場合）、および TCP プロファイルで指定した最大再送数を超えた場合。 • クライアントがサーバの応答を受信しないまま、Inbandモニタの **[Response Time]** で指定した時間が過ぎてしまった（設定値が 0 でない場合）。 • **[Response Time]** で指定した時間に到達する前に接続が切断された。 Inbandモニタによってプールメンバが**down**とマーク付けされてから、**[Retry Time]** 設定で指定した秒数が経過すると（設定値が 0 でない場合）、そのプールメンバは要求を受信できるようになります。クライアントの要求に問題なく応答した時点で、そのプールメンバは **up** とマーク付けされます。

***表 A.1** バーチャルサーバタイプごとの Inband モニタの動作の違い（続き）*

CMP に関する注意事項

BIG-IP で CMP（クラスタリングマルチプロセッシング）機能が有効になっている場合、Inband モニタで指定した [Failure Interval] 内で何度接続に失敗すればプールメンバが **down** とマーク付けされるかは、各 **tmm** インスタンスにおいて算出されます。

たとえば、Inband モニタの **[Failures]** 値を **3**、**[Failure Interval]** 値を **30**（いずれもデフォルト値）と設定した場合、何度失敗すればプールメンバが **down** とマーク付けされるかは、そのシステムで CMP が有効になっているかどうかによって変わってきます。

- CMP が無効の場合（つまり、**tmm** の 1 つのインスタンスのみが動作している）、30 秒以内に受動的な障害が 3 度発生すれば、モニタはそのプールメンバを **down** 状態であると報告します。
- CMP が有効である場合（たとえば、**tmm** の 2 つのインスタンスが動作している）、**tmm** インスタンスのいずれか 1 つで受動的な障害が 3 度発生すれば、モニタはそのプールメンバを **down** 状態であると報告します。しかし、クライアントリクエストが Round Robin 方式で 2 つの **tmm** インスタンス間に分散される場合は、クライアントリクエストに6回失敗した時点で、モニタはプールメンバを**down** 状態であると報告します。

LDAP

LDAP タイプのモニタでは、Lightweight Directory Access Protocol（LDAP）サーバの状態がチェックされます。LDAP プロトコルは、E メールディレクトリの統合のために標準的な X.500 を実装します。指定された基準とフィルタに対してエントリが戻されると、チェックは成功です。LDAP モニタではユーザ名、**パスワード**、および基準文字列とフィルタ文字列が必要です。図 A.11 は、LDAP タイプのモニタの設定とデフォルト値を示します。

```
Name ""
Type LDAP
Interval 10
Timeout 31
Manual Resume No
Check Until Up No
User Name ""
Password ""
Base ""
Filter ""
Security None
Mandatory Attributes No
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図 A.11 LDAP タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_ldap_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 10 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 31 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **User Name** と **Password**
  [**User Name**] 設定では、識別名、つまり LDAP 形式のユーザ名を指定します。
- **Base**
  [**Base**] 設定では、クエリを開始する LDAP 階層の開始地点を指定します。
- **Filter**
  [**Filter**] 設定では、検索項目の LDAP 形式のキーを指定します。
- **Security**
  [**Security**] 設定では、使用するセキュリティプロトコルを指定します。指定可能な値は、**SSL**、**TLS**、または **None** です。
- **Mandatory Attributes**
  [**Mandatory Attributes**] 設定は、BIG-IP がフィルタ検索を実行する方法に作用します。この値が [**No**] に設定されている場合、BIG-IP は属性について 1 レベル検索を実行し、属性が戻らない場合、このノードは **up** として報告されます。この値が [**Yes**] に設定されている場合、BIG-IP はサブツリー検索を実行し、属性が戻らない場合、このノードは **up** としては報告されません。
- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes と設定すると、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

LDAP モニタが適切に動作するには、BIG-IP が LDAP または LDAPS ノードのアドレス上で逆引きDNS参照を実行できる必要があります。この逆引き参照によって、BIG-IP は SSL 証明書の検証時にノードアドレスのホスト名をチェックできます。外部の DNS サーバはこのタイプのモニタとは連動できません。

逆引き DNS 参照の要件は、LDAP が SSL 証明書の使用を必要としない場合でも LDAP ノードと LDAPS ノードの両方に適用されます。

◆注

*各 LDAP サーバのエントリを **/etc/hosts** ファイルに挿入する必要はありません。*

Module Score

Module Score タイプのモニタは、Web Accelerator モジュールと Application Security Manager モジュールに関連付けられた LTM バーチャルサーバに均等に負荷を分散します。Module Score タイプのモニタの設定では、LTM 管理システムが SNMP を利用してダウンストリームのバーチャルサーバから **gtm_score** 値を取り出し、関連するアップストリームの LTM プールメンバやノードの Dynamic Ratio を設定します。

具体的に言うと、Module Score モニタがバーチャルサーバの **gtm_score** 値とそのバーチャルサーバに関連付けられた **gtm_vs_score** 値を取得します。続いて、プール名が指定されていない場合は、モニタはそのバーチャルサーバに関連付けられたノードの Dynamic Ratio を設定します。

BIG-IP では、0 以外のもっとも小さな **gtm_vs_score** 値を使って、Dynamic Ratio を設定します。すべての **gtm_vs_score** 値が 0 の場合は、Dynamic Ratio の設定に **gtm_score** 値が使われます。モニタの定義でプール名を指定すると、そのプールメンバに対して Dynamic Ratio が設定されます。図 A.12 では、Module Score モニタの設定とデフォルト値を示しています。

```
Name ""
Type Module Score
Interval 10
Timeout 30
Manual Resume No
SNMP Community public
SNMP Version v2c
SNMP IP Address ""
SNMP Port 161
Pool Name ""
```

図 A.12 Module Score タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_module_score_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 10 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 30 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。
- **Manual Resume**
  [Manual Resume] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **SNMP IP Address**
  Module Score モニタの設定では、[**Name**] と [**SNMP IP Address**] の値を明示的に指定する必要があります。
- **Pool Name**
  プールメンバに関連付けられたノードではなく、そのプールメンバに対して Dynamic Ratio を設定したいという場合には、プール名のみを指定します。

◆注

WebAccelerator または Application Security Manager のバーチャルサーバのクラスタにトラフィックを分散したい場合には、バックエンド LTM システムごとに 1 つずつカスタムの Module Score モニタを作成する必要があります。

MSSQL

MSSQL タイプのモニタを使用すると、Microsoft® SQL Server Version 6.5/7.0 などの Microsoft® SQL Server ベースのサービスに関するサービスチェックを実行できます。

BIG-IP では、実際のログインを実行する前に JDBC ドライバをインストールする必要があります。詳細については、付録 B「*モニタについての追加留意事項*」を参照してください。

接続が拒否されたことを示すメッセージが表示されたら、IP アドレスとポート番号、またはサービスが正しいことを確認してください。これらが正しくてもログオンできない場合は、第 12 章「*モニタの設定*」を参照してください。

本項の MSSQL モニタに関する後半部分では、前提条件のタスク、デフォルトのモニタ設定、およびトラブルシューティングのヒントについて説明します。

MSSQL の前提条件のタスク

MSSQL タイプのモニタを使用する前に、JDBC Java™ Archive（JAR）ファイル群をダウンロードし、これらを BIG-IP にインストールする必要があります。詳細については、付録第 B 章「*モニタについての追加留意事項*」を参照してください。

MSSQL モニタの設定とそのデフォルト値

図 A.13 は、MSSQL タイプのモニタの設定とデフォルト値を示します。

```
Name ""
Type mssql
Interval 30
Timeout 91
Manual Resume No
Check Until Up No
Send String ""
Receive String ""
User Name ""
Password ""
Database ""
Receive Row ""
Receive Column ""
Count 0
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

***図 A.13** MSSQL タイプのカスタムモニタとデフォルト値*

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_mssql_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 30 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 91 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Send String**
  [**Send String**] はオプション設定であり、BIG-IP がサーバに送信する SQL クエリステートメントを指定します。たとえば、**SELECT * FROM sales** や**SELECT FirstName, LastName From Employees**などです。[**Send String**] を設定する場合、[**Receive String**]、[**Receive Row**]、[**Receive Column**] もあわせて設定することができます。
- **Receive String**
  [**Receive String**] 設定は、[**Receive Row**] 設定と [**Receive Column**] 設定で指定した行とカラムに対する値を指定するオプションパラメータです。たとえば、**Receive String** 値として **ALAN SMITH** を指定します。この設定は、**Send String** 設定を構成している場合だけ行えます。
- **Database**
  [**Database**] 設定では、Microsoft® SQL ベースサーバのデータソースの名前を指定します。たとえば、**sales** や **hr** などです。
- **Receive Row**
  [**Receive Row**] はオプション設定であり、[**Receive String**] 設定が指定されている場合にのみ有効です。この設定では、**Receive String** 値を含む戻りテーブル内に行が指定されます。[**Send String**] と [**Receive String**] を設定する場合にかぎり、この値を指定することができます。
- **Receive Column**
  [**Receive Column**] はオプション設定であり、[**Receive String**] 設定が指定されている場合にのみ有効です。

この設定では、[**Receive String**] 値を含む戻りテーブル内にカラムが指定されます。[**Send String**] と [**Receive String**] を設定する場合にかぎり、この値を指定することができます。

- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。**Yes** と設定すると、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

MSSQL ログインのトラブルシューティング

MSSQL モニタがサーバにログインできず、指定された IP アドレスとポート番号またはサービスが正しいことが確認された場合は、次の操作を試してください。

- **別のツールを使用してログインできることを検証します。**
  たとえば、サーバプログラムの Microsoft Windows® NT® SQL Server version 6.5 に ISQL/w という名前のクライアントプログラムが含まれているとします。このクライアントプログラムでは、SQL サーバへの単純なログインが実行されます。このプログラムを使用して、ISQL/wプログラムを使ってサーバにログインできるかどうかをテストしてください。

- **Microsoft® SQL Enterprise Manager を使用してログオンアカウントを追加します。**
  Microsoft® SQL Server 上で、SQL Enterprise Manager を実行してログインアカウントを追加できます。最初に SQL Enterprise Manager を入力すると、管理する SQL サーバを入力するように求めるメッセージが表示される場合があります。

  マシン名、ユーザ名、およびパスワードを入力してサーバを登録できます。これらの名前が正しい場合、サーバは登録され、サーバのアイコンをクリックできるようになります。サーバのサブツリーを展開すると、ログインアカウントのアイコンが表示されます。

  このサブツリーの下に SQL ログインがあります。ここで、[**Logins**] アイコンを右クリックすると、パスワードを変更したり、新しいログオンを追加したりできます。[**Add login**] オプションにアクセスするにはこのアイコンをクリックしてください。このオプションを開いた後、新しいログインのユーザ名とパスワード、およびこのログインでどのデータベースにアクセスできるかを入力します。EAV 設定で指定するデータベースに対して **test** アカウントアクセス権を付与する必要があります。

NNTP

Usenet News トラフィックの状態をチェックするには、NNTP タイプのモニタを使用します。モニタがサーバからニュースグループ識別ラインを検索すると、このチェックは正常終了しています。NNTP モニタではニュースグループ名（**alt.cars.mercedes** など）、および必要に応じてユーザ名とパスワードが必要です。

図 A.14 は、NNTP タイプのモニタの設定とデフォルト値を示します。

```
Name ""
Type NNTP
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
User Name ""
Password ""
Newsgroup ""
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図 A.14 NNTP タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_nntp_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルト値は **5**（秒単位）です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルト値は **16**（秒単位）です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes の場合、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

Oracle

Oracle タイプのモニタを使用すると、Oracle データベースサーバの状態をチェックできます。モニタがサーバに接続でき、指定されたユーザとしてログインおよびログアウトできれば、このチェックは正常に終了しています。

図 A.15 は、Oracle タイプの**モニタ**の設定とデフォルト値を示します。

```
Name ""
Type Oracle
Interval 30
Timeout 91
Manual Resume No
Check Until Up No
Send String GET /
Receive String ""
User Name ""
Password ""
Database ""
Receive Row ""
Receive Column ""
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図 A.15 Oracle タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_oracle_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 30 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 91 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Send String**
  [**Send String**] 設定は、BIG-IP が Oracle サーバに送信する SQL ステートメントを指定します。たとえば、**SELECT * FROM sales** です。
- **Receive String**
  Receive String 設定は、**Send String** 設定の検索でテーブルの行とカラムに返される値を指定するオプションパラメータです。たとえば、**Receive String** 値として **SMITH** を指定します。
- **Database**
  Oracle タイプのモニタでは、Database の設定によって、Oracle サーバ上のデータソース名が指定されます。たとえば、**sales** や **hr** などです。
- **Receive Row**
  [**Receive Row**] はオプション設定であり、[**Receive String**] 設定が指定されている場合にのみ有効です。この設定では、**Receive String** 値を含む戻りテーブル内に行が指定されます。
- **Receive Column**
  [**Receive Column**] はオプション設定であり、[**Receive String**] 設定が指定されている場合にのみ有効です。この設定では、[**Receive String**] 値を含む戻りテーブル内にカラムが指定されます。
- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes と設定すると、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

POP3

POP3 タイプのモニタでは、Post Office Protocol（POP）トラフィックの状態がチェックされます。モニタがサーバに接続でき、指定されたユーザとしてログインおよびログアウトできれば、このチェックは正常に終了しています。POP3 モニタでは、**ユーザ名**と**パスワード**を指定する必要があります。

図 A.16 は、POP3 タイプの**モニタ**の設定とデフォルト値を示します。

```
Name ""
Type POP3
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
User Name ""
Password ""
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図 A.16 POP3 タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_pop3_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes の場合、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

RADIUS

RADIUS タイプのモニタを使用すると、Remote Access Dial-in User Service（RADIUS）サーバの状態をチェックできます。サーバが要求ユーザを認証すれば、このチェックは正常に終了しています。RADIUS モニタでは、コード番号の**ユーザ名**、**パスワード**、および共有秘密文字列が必要です。

◆注

通常、RADIUS モニタによってチェックされるサーバでは、ハイレベルなセキュリティを維持しながらモニタの認証も可能にするために特別な構成が必要です。

図A.17は、RADIUSタイプのモニタの設定とデフォルト 値を示します。

```
Name ""
Type RADIUS
Interval 10
Timeout 31
Manual Resume No
Check Until Up No
User Name ""
Password ""
Secret ""
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図A.17 RADIUS タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_radius_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 10 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 31 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルス

チェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes と設定すると、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

RPC

BIG-IP を利用して、RPC モニタ経由で、リモートプロシージャコール（RPC）サーバに搭載されたプログラムのアベイラビリティをチェックすることができます。このモニタは、**rpcinfo** コマンドを実行して RPC サーバに対してクエリを発行し、任意のプログラムのアベイラビリティを確認します。

RPC モニタには、**program** と **version** という 2 つの一意の設定が含まれます。[program] 設定では、利用可能なことを確認する必要があるプログラムやアプリケーションを指定します。[version] はオプション設定であり、そのプログラムの具体的なバージョン番号を指定します。バージョン番号を指定しない場合、モニタは **rpcinfo** コマンドを使って、そのプログラムの少なくとも 1 つのバージョンが利用可能なことを確認します。また、RPC モニタの [**mode**] オプションを使えば、TCP または UDP を使用している RPC サーバのアベイラビリティを確認することができます。

モニタのクエリに対して **ready and waiting** ステートメントが含まれる応答が返されれば、そのモニタがチェックしているリソースは利用可能であるということです。

図 A.18 では、RPC タイプのモニタの設定とデフォルト値を示しています。

```
Name ""
Type RPC
Import Settings rpc
Interval 10
Timeout 31
Manual Resume No
Check Until Up No
Mode TCP
Program <name>
Version Number <number>
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図 A.18 RPC タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_rpc_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 10 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 31 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。

- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Mode**
  RPC モニタの [**Mode**] オプションを使えば、TCP または UDP を使用している RPC サーバのアベイラビリティを確認することができます。
- **Program**
  [**Program**] 設定では、利用可能なことを確認する必要があるプログラムやアプリケーションを指定します。
- **Version Number**
  [**Version Number**] は、そのプログラムの具体的なバージョン番号を指定するオプション設定です。バージョン番号を指定しない場合、モニタは **rpcinfo** コマンドを使って、そのプログラムの少なくとも 1 つのバージョンが利用可能なことを確認します。
- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。
- [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes の場合、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

Real Server

Real Server タイプのモニタでは、RealSystem Server データコレクション エージェントを実行するプール、プールメンバ、またはノードのパフォーマンスをチェックできます。チェック後、モニタはそれに応じてトラフィックをダイナミックにロードバランスします。パフォーマンスモニタは、一般に Dynamic Ratio ロードバランシングで使用されます。パフォーマンスモニタと Dynamic Ratio ロードバランシングの詳細については、第 4 章「*ロードバランシングプールの設定*」と付録 B「*モニタについての追加留意事項*」を参照してください。

◆注

ヘルスモニタとは異なり、パフォーマンスモニタではプール、プールメンバ、またはノードの状態は報告されません。

BIG-IP には、**real_server** という設定済みの Real Server モニタが用意されています。図 A.19 は、**real_server** モニタの設定とデフォルト値を示します。

```
Name ""
Type Real Server
Interval 5
Timeout 16
Manual Resume No
Method GET
Command GetServerStats
Metrics ServerBandwidth:1.5, CPUPercentUsage, MemoryUsage,
TotalClientCount
Agent Mozilla/4.0 (compatible: MSIE 5.0; Windows NT)
```

図 A.19 real_server 設定済みモニタ

すべての設定済みモニタと同様、**real_server** モニタはユーザによる変更はできません。ただし、**Metrics** 設定を変更する場合は、カスタムの Real Server モニタを作成し、これにメトリックを追加し、メトリック値を変更できます。

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_real_server_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。

- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。

◆注

*カスタムの Real Server モニタを作成する場合、**Method**、**Command**、および**Agent** 設定の値は変更できません。*

表 A.2 は、すべてのサーバ固有のメトリック、および **GetServerStats** コマンドに適用されるメトリック設定のデフォルト値を示します。

メトリック	デフォルト係数	デフォルトしきい値
ServerBandwidth (Kbps)	1.0	10,000
CPUPercentUsage	1.0	80
MemoryUsage (Kb)	1.0	100,000
TotalClientCount	1.0	1,000
RTSPClientCount	1.0	500
HTTPClientCount	1.0	500
PNAClientCount	1.0	500
UDPTransportCount	1.0	500
TCPTransportCount	1.0	500
MulticastTransportCount	1.0	500

***表 A.2** Real Server モニタのメトリック*

メトリック係数は、メトリックの値が重み付け全体の計算でどの程度重視されるかを決定する因子です。**メトリックのしきい値**は、メトリックに重みを置く場合にメトリックに許可される最大値です。これらの値の使用方法を理解するには、全体的な重み付けを計算する方法を理解する必要があります。全体的な重み付けとは、メトリックごとに算出される相対的な重みの合計です。一方、相対的な重みは次の 3 つの因子に基づきます。

- モニタによって返されるメトリックの値
- 係数値
- しきい値

これらの値を前提として、相対的な重みは次のように計算されます。

w=((しきい値 – 値) / しきい値) * 係数

係数が高いほど、そのメトリックに計算される相対的な重みが大きくなることがわかります。同様に、しきい値が高ければ高いほど、しきい値未満のメトリック値に対する相対的重みの値は大きくなります（値がしきい値に達すると、重みはゼロになります）。

表 A.2 に示されたデフォルト係数とデフォルトしきい値は、*メトリックのデフォルト*であって、*モニタのデフォルト*ではありません。カスタムの **real_server** モニタでユーザ指定の値がモニタのデフォルトよりも優先されるのと同じように、モニタのデフォルトは、メトリックのデフォルトよりも優先されます。たとえば、上記のモニタは **ServerBandwidth** に対して係数値 **1.5** を指定し、その他のメトリックには値を指定しません。これは、モニタが **ServerBandwidth** 係数に対してモニタのデフォルト値である **1.5** を使用し、その他のすべてのメトリックの係数に対してメトリックのデフォルトである**1**を使用することを意味します。ただし、カスタムモニタの **my_real_server** が **ServerBandwidth** 係数として **2.0** を指定するように設定されていた場合、このユーザ指定の値はモニタのデフォルトを上書きします。

モニタデフォルトと異なるのは、メトリック係数としきい値のみです。モニタ内にないメトリックをカスタムモニタに追加する場合、このメトリックは**Metrics**設定のメトリックリストに追加する必要があります。非デフォルト係数またはしきい値を指定するための構文は次のとおりです。

< メトリック >:< 係数 |<*>:< しきい値 >

SASP

一部のネットワークリソース用に IBM® Group Workload Manager を採用している場合は、SASP モニタを使ってそれらのリソースのアベイラビリティを確認することができます。このモニタは、SASP（Server/Application State Protocol）を使って Group Workload Manager と交信するようになっており、Group Workload Manager に対して、管理対象リソースの現在の重要度を問い合わせます。これらの重要度によって、現在どのリソースが最高のレスポンスタイムを提供するのかが決まってきます。モニタは Group Workload Manager（GWM）から情報を受け取ると、リソースの Dynamic Ratio オプションを設定するようになっており、BIG-IP は最適なリソースを選択して接続要求に応答することが可能です。

◆注

SASP モニタを割り当てた場合、リソースは最初に down とマーク付けされます。このようなステータスの変更が生じるのは、GWM がまだリソースに関する情報を持っていないためです。モニタがクエリ結果を受信した時点で、必要に応じてステータスが変更されます。ほとんどの場合、モニタはこれらの結果を数秒以内に受信するようになっています。

図 A.20 では、SASP タイプのモニタの設定とデフォルト値を示しています。

```
Name ""
Type SASP
Import Settings sasp
GWM Interval Automatic
GWM Address 10.10.5.23
GWM Service Port 3000
GWM Protocol TCP
```

***図 A.20** SASP タイプのカスタムモニタとデフォルト値*

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_sasp_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **GWM Interval**
  [**GWM Interval**] オプションは、モニタが GWM にクエリを送信する頻度を指定します。このオプションは、**Automatic**（GWM が推奨するインターバルを使用するようモニタに指示）か、**Specify**（インターバルを指定可能）のどちらかに設定することができます。**Specify** を選択した場合、インターバルは 10 ～ 600 秒の間で指定することが可能です。
- **GWM Address**
  [**GWM Address**] オプションは、Group Workload Manager の IP アドレスを指定します。

- **GWM Service Port**
  [GWM Service Port] オプションでは、SASP モニタが Group Workload Manager との通信に使用するポートを指定します。
- **GWM Protocol**
  [GWM Protocol] オプションでは、モニタが使用するプロトコル（TCP または UDP）を指定することができます。

Scripted

作成するファイルを読み取る簡単なスクリプトを生成するには、Scripted タイプのモニタを使用します。ファイルには、送信する行または受信する行を指定する **send** 文字列と **expect** 文字列が含まれます。たとえば、図 A.21 は単純な SMTP シーケンスを指定する、作成可能なサンプルファイルを示します。ファイルの行は、常に指定された順序で読み取られます。

```
expect 220
send "HELO bigip1.somecompany.net\r\n"
expect "250"
send "quit\r\n
```

***図 A.21** Scripted モニタの SMTP シーケンスファイルの例*

Scripted モニタを使用すると、上記のファイル上で動作するスクリプトを生成できます。Scripted モニタスクリプトはこのファイルを読み取ると、各行を調べ、行に引用符がない場合、その行は送信されるか、そのまま受信されることになります。行が引用符で囲まれている場合、引用符がはずされ、その行にエスケープ文字がないかが調べられ、それに応じて処理されます。

図A.22は、Scriptedタイプのモニタの設定とデフォルト値を示します。.

```
Name ""
Type Scripted
Import Settings scripted
Interval 10
Timeout 31
Manual Resume No
Check Until Up No
Filename <filename>
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

***図 A.22** Scripted タイプのカスタムモニタとデフォルト値*

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_scripted_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 10 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 31 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。

- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Filename**
  [**Filename**] 設定では、作成するファイルの名前を指定します。ファイルには、送信する行または受信する行を指定する **send** 文字列と **expect** 文字列が含まれます。
- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。
- [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes と設定すると、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

◆注

*送信文字列と受信文字列を含むファイルを作成したら、このファイルをディレクトリ **/config/eav** に保存してください。*

SIP

SIP サービスの状態をチェックするには、SIP タイプのモニタを使用します。デフォルトでは、このタイプのモニタはサーバデバイスに対して UDP で SIP OPTIONS 要求を発行するようになっていますが、別のプロトコル（TCP、TLS、SIPS（Secure SIP））を使用することも可能です。

SIP モニタがデバイスに対して発行する要求は、サーバデバイスがサポートするオプションを識別することを目的としたものです。適切な要求が返されると、デバイスは **up** していてコマンドに応答していると見なされます。

図 A.23 は、SIP タイプのモニタの設定とデフォルト値を示します。

```
Name ""
Type SIP
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Mode UDP
Cipher List DEFAULT:+SHA:+3DES:+kEDH
Compatibility Enabled
Client Certificate None
Client Key None
Additional Accepted Status Codes None
Additional Rejected Status Codes None
Header List ""
SIP Request ""
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図 A.23 SIP タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_sip_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使って、リソースが有効であると手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Mode**
  [**Mode**] に指定できる値は、**TCP**、**UDP**、**TLS**、**SIPS** です。
- **Cipher List**
  このモニタでは、**TLS** モードと **SIPS** モードの場合にかぎり、暗号リストを指定します。
- **Compatibility**
  TLS モードと **SIPS** モードの場合にかぎり、SSL が有効になっていれば、OpenSSL の SSL オプションを **ALL** に設定します。
- **Client Certificate**
  TLS モードと **SIPS** モードの場合にかぎり、モニタがターゲットの SSL サーバに送信するクライアント証明書を指定します。
- **Client Key**
  TLS モードと **SIPS** モードの場合にかぎり、モニタがターゲットの SSL サーバに送信するクライアント証明書のキーを指定します。
- **Additional Accepted Status Codes**
  [**Additional Accepted Status Codes**] 設定に指定できる値は、**Any**、**None**、および **Status Code List** です。**Status Code List** を選択すると、状態コード **200** のほか、システムによる扱いが異なる複数の許容状態コードを指定することができます。複数の状態コードはスペースで区切る必要があります。**Any** を指定すると、サーバを **up** とマーク付けするのに、すべての状態コードが使用されることになります。この設定と [**Additional Rejected Status Codes**] 設定の値が同じである場合は、[**Additional Accepted Status Codes**] 設定が優先されます。
- **Additional Rejected Status Codes**
  [**Additional Rejected Status Codes**] 設定に指定できる値は、**Any**、**None**、および **Status Code List** です。**Status Code List** を選択した場合、システムによる扱いが異なる複数の拒絶状態コードを指定することができます。複数の状態コードはスペースで区切る必要があります。**Any** を指定すると、サーバを **down** とマーク付けするのに、すべての状態コードが使用されることになります。この設定と [**Additional Accepted Status Codes**] 設定の値が同じである場合は、[**Additional Accepted Status Codes**] 設定が優先されます。
- **Header List**
  [**Header List**] 設定を使うことで、ヘッダの追加、編集、削除が行えます。ただし、SIP モニタが使用するデフォルトヘッダについては削除できません。また、いくつかのヘッダでは、ヘッダ全体をオーバーライドする以外、デフォルトヘッダを補強する（パラメータを追加するなど）ことはできなくなっています。
- **SIP Request**
  [**SIP Request**] 設定では、SIP メッセージの要求行を指定します。その際かならず、末尾の **\r\n** を除く SIP 要求行全体を指定するようにしてください。システムは、この要求に対するレスポンスコード

を使って、サーバの up/down を特定します。モニタは SIP サーバに対して、カスタマイズされたシンプルなクエリを実行しますが、接続の確立、ハンドシェイキングの実行、SIP トラフィックや要求の処理には関与しません。サーバに要求を送信し、レスポンスコードを確認するだけで、（応答と要求を照合する以外）応答の残りの部分については無視します。結果、このモニタはダイアログには参加せず、INVITE などの要求をサポートすることはありません。

- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。**Yes** と設定すると、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

SMB

SMB モニタでは、BIG-IP を使って SMB/CIFS（Server Message Block/Common Internet File System）サーバのアベイラビリティをチェックすることができます。このモニタを採用することで、サーバ全体のアベイラビリティ、サーバの特定のサービスのアベイラビリティ、もしくはサービスが使用する特定のファイルのアベイラビリティのチェックが可能になります。

サーバで [**SMB/CIFS Server**] 設定を使用するように指定するだけで、[**Service Name**] と [**Path/Filename**] を空欄のままにしておくと、モニタはSMB/CIFSサーバから使用可能なサービスのリストを取得しようと試みます。モニタがリストを取得した場合、そのサーバは **available** とマーク付けされます。

図 A.24 では、SMB タイプのモニタの設定とデフォルト値を示しています。

```
Name ""
Type SMB
Import Settings smb
Interval 10
Timeout 31
Manual Resume No
Check Until Up No
User Name <name>
Password <password>
Path/Filename <filename and path>
SMB/CIFS Server <name>
Service Name <name>
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図 A.24 SMB タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_smb_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 10 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 31 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Check Until Up**
  [Check Until Up] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Path/Filename**
  サービスに関連付けられたファイルを指定します。モニタは、サービスそのものに対する相対パスを使って、ファイルの検索を行います。必ずしも値を指定する必要はありませんが、このオプションの使用を選択した場合、**[Service Name]** 設定によりサービスをあわせて指定しておく必要があります。

- **SMB/CIFS Server**
  モニタがアベイラビリティのチェックを行う SMB/CIFS サーバの NetBIOS 名を指定します。このモニタを機能させるにはサーバを指定する必要があります。

- **Service Name**
  SMB/CIFS のどのサービスのアベイラビリティを確認したいのかを指定します。サービス名は、必ずしも指定する必要はありません。

- **Alias Address** と **Alias Service Port**
  [Alias Address] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Debug**
  そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。**[Debug]** に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes と設定すると、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

SMTP

SMTP タイプのモニタでは、Simple Mail Transport Protocol（SMTP）サーバの状態がチェックされます。このタイプのモニタは、サーバが **up** してコマンドに対応していることのみをチェックする、極めて基本的なモニタです。メールサーバが標準 SMTP **HELO** コマンドと **QUIT** コマンドに応答すると、このチェックは成功です。

図 A.25 は、SMTP タイプのモニタの設定とデフォルト値を示します。

```
Name ""
Type SMTP
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Domain ""
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図 A.25 SMTP タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_smtp_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Domain**
  SMTPタイプのモニタでは、ドメイン名を指定する必要があります。

- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes と設定すると、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

SNMP DCA

SNMP DCA タイプのモニタでは、UC Davis などの SNMP エージェントを実行するサーバのパフォーマンスをチェックし、それに応じてサーバのトラフィックをロードバランスできます。このモニタを使用すると、CPU、メモリ、およびディスク使用の重み付けを定義できます。

パフォーマンスモニタは、一般に Dynamic Ratio ロードバランシングで使用されます。パフォーマンスモニタと Dynamic Ratio ロードバランシングの詳細については、第 4 章「*ロードバランシングプールの設定*」と付録 B「*モニタについての追加留意事項*」を参照してください。

◆注

ヘルスモニタとは異なり、パフォーマンスモニタではプール、プールメンバ、またはノードの状態は報告されません。

BIG-IP には、**snmp_dca** という設定済みの SNMP DCA モニタが用意されています。図 A.26 は、設定済みモニタ **snmp_dca** の設定と値を示します。

```
Name ""
Type SNMP DCA
Interval 10
Timeout 30
Community Public
Version v1
Agent Type UCD
CPU Coefficient 1.5
CPU Threshold 80
Memory Coefficient 1.0
Memory Threshold 70
Disk Coefficient 2.0
Disk Threshold 90
Variables ""
```

図 A.26 snmp_dca 設定済みモニタ

設定済みモニタはユーザによる変更はできません。したがって、SNMP DCA モニタ設定の値を変更する場合は、SNMP DCA タイプのカスタムモニタを作成する必要があります。**Version** 設定に指定できる値は、**v1**、**v2c**、および **Other** です。**Agent Type** 設定に指定できる値は、**UCD**、**Win2000**、および **Other** です。

SNMP DCA カスタムモニタを設定すると、モニタに指定されているデフォルトの CPU、メモリ、およびディスク係数としきい値を使用できます。また、このデフォルトの値を変更することもできます。オプションで、その他の種類のデータを収集する係数としきい値を指定できます。設定するモニタが UC Davis 以外の SNMP エージェントタイプ向けの場合は、**Win2000** などのエージェントタイプを指定する必要があります。

係数としきい値の使用方法を理解するには、全体的な重み付けを計算する方法について把握しておく必要があります。全体的な重み付けとは、メトリックごとに算出される相対的な重みの合計です。一方、相対的な重みは次の 3 つの因子に基づきます。

- モニタによって返されるメトリックの値
- 係数値

- しきい値

これらの値を前提として、相対的な重みは次のように計算されます。

w=((しきい値 – 値)/ しきい値)* 係数

係数が高いほど、そのメトリックに計算される相対的な重みが大きくなることがわかります。同様に、しきい値が高ければ高いほど、しきい値未満のメトリック値に対する相対的重みの値は大きくなります(値がしきい値に達すると、重みはゼロになります)。

SNMP DCA Base

UC Davis などの SNMP エージェントを実行しているサーバのパフォーマンスをチェックするには、SNMP DCA Base タイプのモニタを使用します。ただし、このモニタは宛先を CPU、メモリ、またはディスク使用ではなく、ユーザデータ使用のみに基づいてロードバランスする場合にのみ使用してください。

図 A.27 は、SNMP DCA Base タイプのモニタの設定とデフォルト値を示します。

```
Name ""
Type snmp_dca_base
Interval 10
Timeout 30
Community Public
Version v1
Variables ""
```

図 A.27 SNMP DCA タイプのカスタムモニタとデフォルト値

パフォーマンスモニタは、一般に Dynamic Ratio ロードバランシングで使用されます。パフォーマンスモニタと Dynamic Ratio ロードバランシングの詳細については、第 4 章「*ロードバランシングプールの設定*」と付録第 B 章「*モニタについての追加留意事項*」を参照してください。

◆注

ヘルスモニタとは異なり、パフォーマンスモニタではプール、プールメンバ、またはノードの状態は報告されません。

SOAP

SOAP モニタは Simple Object Access Protocol（SOAP）に基づいて Web サービスをテストします。より具体的に言うと、このモニタは SOAP ベースの Web サービスに要求をサブミットし、オプションで戻り値または障害を検証します。図 A.28 は、SOAP タイプのモニタの設定とデフォルト値を示します。

```
Name ""
Type soap
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
User Name ""
Password ""
Protocol HTTP
URL Path ""
Namespace ""
Method ""
Parameter Name ""
Parameter Type bool
Parameter Value ""
Return Type bool
Return Value ""
Expect Fault No
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図 A.28 SOAP タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_soap_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。また、ターゲットの応答に RESET パケットが含まれる場合は、タイムアウトインターバルの設定値に達する前に、そのターゲットに対してただちに **down** のフラグが立てられます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Protocol**
  [**Protocol**] 設定で指定できる値は、**HTTP** と **HTTPS** です。
- **Parameter Type**
  [**Parameter Type**] 設定で指定できる値は、**bool**、**int**、**long**、**string** です。
- **Return Type**
  [**Return Type**] 設定で指定できる値は、**bool**、**int**、**short**、**long**、**float**、**double**、**string** です。
- **Expect Fault**
  [**Expect Fault**] 設定で指定できる値は **No** と **Yes** です。
- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes と設定すると、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

TCP

TCP タイプのモニタは、TCP を介して送信された特定のコンテンツの受信を試みます。コンテンツが **Receive String** 値と一致すればチェックは成功です。図 A.29 は、設定済みモニタ **tcp** の設定を示します。

```
Name ""
Type TCP
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Send String ""
Receive String ""
Reverse No
Transparent No
Alias Address * All Addresses
Alias Service Port * All Ports
```

図 A.29 tcp 設定済みモニタ

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_tcp_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使って、リソースが有効であると手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Send String** と **Receive String**
  このタイプのモニタでは、[**Send String**] 値と [**Receive String**] 値が使われます。[**Send String**] 値が空白で、接続が確立できる場合、サービスは **up** であるとみなされます。空白の [**Receive String**] 値はどのような応答とも一致します。コンテンツが [**ReceiveString**] 値と一致すればチェックは成功です。

- **Transparent** と **Reverse**
  Transparent モードと **Reverse** モードはいずれもオプションです。**Transparent** モードと **Reverse** モードの詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Alias Address** と **Alias Service Port**
  [Alias Address] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

TCP Echo

TCP Echo タイプのモニタでは、Transmission Control Protocol（TCP）接続を検証できます。このチェックは、BIG-IP が TCP Echo メッセージへの応答を受信すると成功です。

TCP Echo モニタタイプを使用するには、監視中のノード上で TCP Echo が有効になっていることを確認する必要があります。図 A.30 は、設定済みモニタ **tcp_echo** の設定を示します。

```
Name ""
Type TCP Echo
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Transparent No
Alias Address * All Addresses
```

図 A.30 tcp_echo 設定済みモニタ

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_tcp_echo_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Transparent**
  Transparent モードはこのタイプのモニタのオプションであり、このモードを [**Yes**] に設定すると、モニタはモニタが関連付けられているノードに ping コマンドを発行します。**Transparent** モードの詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

TCP Half Open

モニタの TCP Half Open タイプは、TCP SYN パケットをサービスに送信することによって、関連付けられているサービスをすばやくチェックします。モニタは SYN-ACK パケットをサービスから受信するとすぐに、サービスが **up** 状態にあるとみなし、3 方向ハンドシェイクを完了する代わりに、サービスに RESET を送信します。

図 A.31 は、設定済みモニタ **tcp_half_open** の設定を示します。

```
Name ""
Type TCP Half Open
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Transparent No
Alias Addresses * All Addresses
Alias Service Port * All Ports
```

図 A.31 tcp_half_open 設定済みモニタ

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_tcp_half_open_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Transparent**
  Transparent モードはこのタイプのモニタのオプションであり、このモードを [**Yes**] に設定すると、モニタはモニタが関連付けられているノードに ping コマンドを発行します。**Transparent** モードの詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Alias Address** と **Alias Service Port**
  [**Alias Address**] 値はモニタがチェックする宛先 IP アドレスを指定するものであり、デフォルト値は *** All Addresses** です。詳細については、第 12 章「*モニタの設定*」を参照してください。

UDP

User Datagram Protocol（UDP）パケットを送信する場合は UDP タイプのモニタを使用してください。UDP サービスの状態をチェックすることを目的とした UDP タイプのモニタは、1 つ以上の UDP パケットをターゲットのプール、プールメンバ、またはノードに送信します。

図 A.32 は、UDP タイプのモニタの設定とデフォルト値を示します。.

```
Name ""
Type UDP
Interval 5
Timeout 16
Manual Resume No
Check Until Up No
Send String default send string
Send Packets 2
Timeout Packets 2
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

図 A.32 UDP タイプのカスタムモニタとデフォルト値

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_udp_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Timeout Packets**
  [**Timeout Packets**] の設定値（秒単位）は、[**Interval**] 設定よりも小さな値である必要があります。

- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes と設定すると、エラーメッセージと追加情報が **/var/log/<monitor_type>_<ip_address>.<port>.log** ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

UDP タイプのモニタを使用してプール、プールメンバ、またはノードを監視する場合は、**ICMP** のようなプール、プールメンバ、またはノードを監視する別のタイプのモニタを有効にする必要があります。UDP タイプのモニタと別のタイプのモニタの両方が UDP サービスの状態を **up** としてレポートするまで、UDP サービスはトラフィックを受信しません。詳細は表 A.3 を参照してください。

UDP モニタがレポートする状態	別のモニタがレポートする状態	UDP サービスの状態
up	up	up
up	down	down
down	up	down
down	down	down

***表 A.3** UDP サービスの状態の決定*

WAP

Wireless Application Protocol（WAP）サーバを監視する場合は WAP モニタを使用してください。WAP モニタの一般的な用途は、**Send String** 設定と **Receive String** 設定を指定することのみです。WAP モニタは、URL を要求し、URL 応答で返されたデータ内にある [**Receive String**] 設定の文字列を検出することによって機能します。図 A.33 は、WAP タイプのモニタの設定とデフォルト値を示します。.

```
Name ""
Type WAP
Import Settings wap
Interval 10
Timeout 31
Manual Resume No
Check Until Up No
Send String ""
Receive String ""
Secret ""
Accounting Node ""
Accounting Port ""
Server ID ""
Call ID ""
Session ID ""
Framed Address ""
Alias Address * All Addresses
Alias Service Port * All Ports
Debug No
```

***図 A.33** WAP タイプのカスタムモニタとデフォルト値*

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_wap_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 10 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 31 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。
- **Manual Resume**
  [**Manual Resume**] 設定を使用して、リソースの有効性を手動で指定することができます。詳細については、第 12 章「*モニタの設定*」を参照してください。
- **Check Until Up**
  [**Check Until Up**] 機能を有効にした場合、モニタはプールメンバが up であることが確定するまで、従来通りにそのメンバのヘルスチェックを行います。

プールメンバが up であることが確定すると、そのメンバに対するヘルスチェック機能は無効になります。詳細については、第 12 章「*モニタの設定*」を参照してください。

- **Send String**
  [**Send String**] 設定では URL を指定します。
- **Receive String**
  [**Receive String**] 設定では、モニタが検出することになる、URL 応答により返されたデータ内の文字列を指定します。
- **Secret**
  [**Secret**] 設定は、クライアントと RADIUS サーバの両方が認識する文字列である RADIUS secret であり、MD5 ハッシュ計算で使用されます。
- **Accounting Node**
  [**Accounting Node**] 設定では RADIUS ノードを指定します。この値が Null 文字列であり、RADIUS アカウンティングが要求されている場合（アカウンティングポートは非ゼロ）、WAP サーバノードも同じく RADIUS ノードであると想定されます。
- **Accounting Port**
  [**Accounting Port**] 設定では RADIUS アカウンティングを実装します（オプション）。RADIUS アカウンティングを実装する場合は、アカウンティングポートを非ゼロの値に設定する必要があります。この場合、モニタは RADIUS アカウンティングが必要であるとみなし、指定されたアカウンティングノードとポートにアカウンティング要求が送信され、アカウンティングが開始されます。これは URL が要求される前に実行されます。URL の検索が正しいデータで正常に終了すると、アカウンティング要求が送信されてアカウンティングが停止されます。
- **Server ID**
  Server ID 設定は、要求サーバ（つまり BIG-IP）の RADIUS NAS-ID を指定します。これは FQDN のエイリアスとして使用される文字列です。
- **Call ID**
  Call ID 設定は、電話番号に似た識別子です。つまり、数字の文字列です。テスト目的では、通常この値は 11 文字の文字列です。
- **セッション ID**
  Session ID 設定は、このセッションの識別に使用される RADIUS セッション ID です。これは多くの場合、**01234567** のような任意の数字の文字列です。
- **Framed Address**
  Framed Address 設定は、RADIUS フレーム IP アドレスです。この設定には特殊な用途はなく、通常は単純に **1.1.1.1** と指定されます。
- **Debug**
  [**Debug**] 設定では、そのモニタ用に作成されラベル付けされたログファイルに、エラーメッセージと追加情報を送信するかどうかを指定します。[**Debug**] に指定できる値は **No** と **Yes** です。デフォルト値は **No** であり、この場合、このモニタに関連するエラーメッセージと追加情報がリダイレクトされることはありません。
  Yes と設定すると、エラーメッセージと追加情報が

/var/log/<monitor_type>_<ip_address>.<port>.log ファイルにリダイレクトされます。これらのログ情報は、失敗したヘルスチェックの診断やトラブルシューティングを行うのに役立ちます。

WMI

WMI タイプのモニタは、Windows Management Infrastructure (WMI) データコレクションエージェントを実行しているプール、プールメンバ、またはノードのパフォーマンスをチェックし、それに応じてトラフィックをダイナミックにロードバランスします。

通常、Dynamic Ratio ロードバランシングを実行する WMI モニタなどのパフォーマンスモニタを使用してください。パフォーマンスモニタと Dynamic Ratio ロードバランシングの詳細については、第 4 章「*ロードバランシングプールの設定*」と付録 B「*モニタについての追加留意事項*」を参照してください。

◆注

ヘルスモニタとは異なり、パフォーマンスモニタではプール、プールメンバ、またはノードの状態は報告されません。

図 A.34 は、WMI タイプのモニタの設定とデフォルト値を示します。

```
Name ""
Type wmi
Interval 5
Timeout 16
User Name ""
Password ""
Method POST
URL /scripts/f5Isapi.dll
Command GetCPUInfo, GetDiskInfo, GetOSInfo
Metrics LoadPercentage, DiskUsage, PhysicalMemoryUsage:1.5,
VirtualMemoryUsage:2.0
Agent Mozilla/4.0 (compatible: MSIE 5.0; Windows NT)
Post RespFormat=HTML
```

図 A.34 WMI タイプのカスタムモニタとデフォルト値

カスタム WMI モニタを作成する場合、変更が必要なデフォルト値は、名前と、ユーザ名、パスワードの Null 値のみです。また、**Method** 設定の値は変更できないため注意が必要です。

これらの設定の詳細は以下のとおりです。

- **Name**
  カスタムモニタの名前を一意で指定します（「**my_wmi_monitor**」など）。
- **Type**
  作成するモニタのタイプを指定します。
- **Interval**
  モニタチェックを行う頻度を指定します。デフォルトは 5 秒です。
- **Timeout**
  モニタの要求にノードが何秒以内に応答しなければならないかを指定します。デフォルトは 16 秒です。設定時間内に応答すれば、そのノードは **up** と見なされ、設定時間内に応答しなければ **down** と見なされます。[**Timeout**] 値は [**Interval**] 値の 3 倍 +1 秒となるように設定します。

- **User Name** と **Password**
  パスワードによるセキュリティがない場合は、[**Username**] 設定と [**Password**] 設定に空白文字列 **""** を使用します。
- **Command** と **Metrics**
  以下の表を参照してください。

表 A.4 は、**Command** 設定と **Metrics** 設定で指定できるすべてのコマンドとメトリックを示します。また、デフォルトのメトリック値も示します。

コマンド	メトリック	デフォルト係数	デフォルトしきい値
GetCPUInfo	LoadPercentage (%)	1.0	80
GetOSInfo	PhysicalMemoryUsage (%)	1.0	80
	VirtualMemoryUsage (%)	1.0	80
	NumberRunningProcesses	1.0	100
GetDiskInfo	DiskUsage (%)	1.0	90
GetPerfCounters	TotalKBytesPerSec	1.0	10,000
	ConnectionAttemptsPerSec	1.0	500
	CurrentConnections	1.0	500
	GETRequestsPerSec	1.0	500
	PUTRequestsPerSec	1.0	500
	POSTRequestsPerSec	1.0	500
	AnonymousUsersPerSec	1.0	500
	CurrentAnonymousUsers	1.0	500
	NonAnonymousUsersPerSec	1.0	500
	CurrentNonAnonymousUser	1.0	500
	CGIRequestsPerSec	1.0	500
	CurrentCGIRequests	1.0	500
	ISAPIRequestsPerSec	1.0	500
	CurrentISAPIRequests	1.0	500
GetWinMediaInfo	AggregateReadRate	1.0	10,000 Kbps
	AggregateSendRate	1.0	10,000 Kbps
	ActiveLiveUnicastStreams	1.0	1000

***表 A.4** WMI タイプのモニタコマンドとメトリック*

コマンド	メトリック	デフォルト係数	デフォルトしきい値
	ActiveStreams	1.0	1000
	ActiveTCPStreams	1.0	1000
	ActiveUDPStreams	1.0	1000
	AllocatedBandwidth	1.0	10,000 Kbps
	AuthenticationRequests	1.0	1000
	AuthenticationsDenied	1.0	100
	AuthorizationRequests	1.0	1000
	AuthorizationsRefused	1.0	100
	ConnectedClients	1.0	500
	ConnectionRate	1.0	500
	HTTPStreams	1.0	1000
	HTTPStreamsReadingHeader	1.0	500
	HTTPStreamsStreamingBody	1.0	500
	LateReads	1.0	100
	PendingConnections	1.0	100
	PluginErrors	1.0	100
	PluginEvents	1.0	100
	SchedulingRate	1.0	100
	StreamErrors	1.0	100
	StreamTerminations	1.0	100
	UDPResendRequests	1.0	100
	UDPResendsSent	1.0	100

***表 A.4** WMI タイプのモニタコマンドとメトリック（続き）*

B

モニタについての追加留意事項

- モニタへの Dynamic Ratio ロードバランシングの実装
- MSSQL モニタの実装

モニタへの Dynamic Ratio ロードバランシングの実装

RealNetworks® RealServer™ サーバ、WMI（Windows Management Instrumentation）を搭載した Microsoft® Windows® サーバ、および SNMP エージェント（UC Davis SNMP エージェント、Windows® 2000 Server SNMP エージェントなど）を搭載したサーバから構成されるプールに、Dynamic Ratio ロードバランシングを設定できます。

Dynamic Ratio ロードバランシングをこれらのタイプのサーバに実装するため、BIG-IP® ローカルトラフィック管理システムでは、各タイプのサーバ用のモニタプラグインファイルと、ヘルスモニタおよびパフォーマンスモニタを装備しています。SNMP エージェントを搭載したサーバは例外です。この場合、BIG-IP はモニタのみを提供します。SNMP エージェントを実行するサーバには特殊なプラグインファイルは必要ありません。

モニタするそれぞれのサーバにモニタプラグインをインストールして、BIG-IP に常駐するパフォーマンスモニタを作成する必要があります。モニタを作成すると、モニタはサーバプラグインと直接通信します。表 B.1 に、各サーバタイプに対して必要なモニタプラグインと、対応するパフォーマンスモニタのタイプを示します。.

サーバタイプ	モニタプラグイン	モニタタイプ
RealServer Windows サーバ	**F5RealMon.dll**	Real Server
RealServer UNIX サーバ	**f5realmon.so**	Real Server
WMI搭載のWindowsサーバ	**F5Isapi.dll または F5.IsHandler.dll**	WMI
Windows 2000 Serverサーバ	SNMP エージェント	SNMP DCA および SNMP DCA Base
UNIX サーバ	UC Davis SNMPエージェント	SNMP DCA および SNMP DCA Base

表 B.1 *モニタプラグインと、対応するモニタテンプレート*

Real Server モニタの実装

RealSystem™ Server の場合、RealSystem Server にプラグインをインストールすると、BIG-IP が必要なメトリックを収集するモニタプラグインを提供します。RealSystem Server で Dynamic Ratio ロードバランシングを設定するには、次の 4 つのタスクを実行します。

- モニタプラグインを RealSystem サーバにインストールする
- Real Server モニタを BIG-IP に設定する
- メトリックを収集するサーバにモニタを関連付ける
- Dynamic Ratio ロードバランシング方式を使用するサーバプールを、作成または変更する

モニタプラグインを RealSystemServer にインストールするには（Windows バージョン）

1. モニタプラグイン **F5RealServerPlugin.dll** を BIG-IP からダウンロードします。
   プラグインは、フォルダ **/usr/local/www/docs/agents** にあります。
2. **F5RealServerPlugin.dll** を RealServer の **plug-ins** ディレクトリにコピーします。
   （例：**C:\Program Files\RealServer\plug-ins**）
3. RealSystem Server プロセスが実行中の場合は、再起動します。

Linux または UNIX RealSystem Server モニタプラグインをインストールまたはコンパイルするには

1. **.iso** イメージを使用して、BIG-IP ソフトウェアの CR-ROM を作成します。
2. CD で、**/downloads/rsplug-ins** ディレクトリに移動します。
3. ファイル **F5RealMon.src.tar.gz** を BIG-IP のディレクトリ **/var/tmp** にコピーします。
4. BIG-IP で、ディレクトリ **/var/tmp** に移動します。
   ```
   cd /var/tmp
   ```
5. UNIX の **tar** コマンドを使用して、**F5RealMon.src.tar.gz** ファイルを展開します。
   ```
   tar -xvzf F5RealMon.src.tar
   ```
6. **F5RealMon.src** ディレクトリに移動します。
   ```
   cd F5RealMon.src
   ```
7. **ls** コマンドを入力して、ディレクトリの内容を表示します。
8. ソースをコンパイルするには、ファイル **build_unix_note** の手順を使用します。
9. RealSystem Server を起動します。

プラグインをインストールしてコンパイルしたら、Real Server モニタを設定して、設定したモニタをプールメンバに関連付け（RealSystem Server サーバ）、ロードバランシング方式として Dynamic Ratio を設定する必要があります。

- Real Server モニタの設定については、第 12 章「*モニタの設定*」を参照してください。
- パフォーマンスモニタのプールメンバへの関連付けについては、第 4 章「*ロードバランシングプールの設定*」を参照してください。
- プールのロードバランシング方式を Dynamic Ratio 方式に設定する方法については、第 4 章「*ロードバランシングプールの設定*」を参照してください。

WMI モニタの実装

WMI（Windows Management Instrumentation）を実行している Windows では、BIG-IP LTM はサーバにデータ収集エージェント（**F5Isapi.dll**（IIS 5.0 と 6.0）、または **F5.IsHandler.dll**（IIS 6.0 と 7.0））を提供します。Windows プラットフォームで Dynamic Ratio ロードバランシングを設定するには、次の 4 つのタスクを実行します。

- IIS サーバにデータ収集エージェントをインストールする。詳細については、この項に記載された手順を参照してください。
- BIG-IP で WMI モニタを設定する。
  WMI モニタの設定については、第 12 章「*モニタの設定*」を参照してください。
- メトリックを収集するサーバにモニタを関連付ける。
  カスタムモニタのプールメンバへの関連付けについては、第 4 章「*ロードバランシングプールの設定*」を参照してください。
- Dynamic Ratio ロードバランシング方式を使用するサーバプールを作成または変更する。
  プールのロードバランシング方式を Dynamic Ratio 方式に設定する方法については、第 4 章「*ロードバランシングプールの設定*」を参照してください。

IIS サーバにデータ収集エージェントをインストールする手順は、サーバが 5.0、6.0、7.0 のどのバージョンの IIS を実行しているかによって、またそのデータ収集エージェントが **F5Isapi.dll** か **F5.IsHandler.dll** のどちらであるかによって変わってきます。データ収集エージェントファイル、およびそれぞれのファイルがサポートされている IIS バージョンについては、表 B.2 を参照してください。

データ収集エージェント	IIS バージョン 5.0	IIS バージョン 6.0	IIS バージョン 7.0
F5Isapi.dll	X	X	
F5.IsHandler.dll		X	X

***表 B.2** 各 IIS バージョンでサポートされるデータ収集エージェントファイル*

*WMI モニタの実装*本項の後半では、**F5Isapi.dll** と **F5.IsHandler.dll** の 2 つのデータ収集エージェントファイルのインストール手順について説明します。

データ収集エージェント F5Isapi.dll のインストール

F5isapi.dll ファイルは、IIS バージョン 5.0 と 6.0 にインストールすることができます。

データ収集エージェント F5Isapi.dll を IIS 5.0 サーバにインストールするには

1. **データ収集エージェント（F5Isapi.dll）**を BIG-IP からダウンロードします。
   このプラグインは、BIG-IP の **/var/windlls** または **/usr/local/www/docs/agents** ディレクトリにあります。
2. **f5isapi.dll** をディレクトリ **C:\Inetpub\scripts** にコピーします。
3. Internet Services Manager を開きます。
4. Internet Services Manager の左側のペインで、フォルダ **<machine_name>\Default Web Site\Script** を開きます。ここで、**<machine_name>** は設定しているサーバの名前です。**Scripts** フォルダの内容は、右側のペインに表示されます。
5. 右側のペインで、**F5Isapi.dll** を右クリックして、[**Properties**] を選択します。
   F5Isapi.dll の [Properties] ダイアログボックスが開きます。
6. [**Logvisits**] の選択を解除します。
   (エージェントへのアクセスをすべてログに記録すると、ログファイルがすぐにいっぱいになります)
7. [**File Security**] タブをクリックします。
   [File Security] オプションが開きます。
8. [**Anonymous access and authentication control group**] ボックスで、[**Edit**] をクリックします。
   [Authentication Methods] ダイアログボックスが開きます。
9. ダイアログボックスで、すべてのチェックボックスをクリアして、[**Basic Authentication**] を選択します。
10. [**Authentication methods**] ダイアログボックスで、[**OK**] をクリックして変更を受理します。
11. [**Properties**] ダイアログボックスで、[**Apply**] をクリックします。
    WMI データ収集エージェントを使用できるようになりました。

データ収集エージェント F5Isapi.dll を IIS 6.0 サーバにインストールするには

1. Web サイトドキュメントのルートで **scripts** ディレクトリを作成します（**Default Website**（デフォルト Web サイト）場合は **C:\InetPub\wwwroot**）。
2. **scripts** ディレクトリのプロパティを、**scripts and executables**（スクリプトと実行可能ファイル）に設定します。
3. **f5isapi.dll** ファイルを、作成した **scripts** ディレクトリにコピーします。
4. IIS Manager（**inetmgr**）を起動し、**scripts** ディレクトリまで移動します。
5. 右側のペインで、ファイル名 **f5isapi.dll** を選択します。

6. [**Properties] -> [File Security] -> [Authentication and Access Control**] を順に選択し、設定 [**anonymous user**] と [**Basic Authentication**] がチェックされていることを確認します。
7. 未知のエクステンションをすべて許可する場合、IIS Manager で、**[Web Server Extensions] -> [All Unknown ISAPI extensions]** まで移動し、すべての未知のエクステンションを有効にします。許可しない場合、ステップ 8 へ進みます。
8. **f5isapi.dll** ファイルだけを許可したい場合、[**Web Server Extensions]-> [Tasks: Add a New Webserver Extension**] の順に選択します。続いて、以下の手順を実行します。
   a) [**Name**] フィールドで、[**F5 ISAPI**] を選択し、必要なファイルで [**Add**] をクリックします。
   これにより、ファイルへのパスを要求します。
   b) **Default Website**（デフォルト Web サイト）の場合はパス **C:\InetPub\wwwroot\scripts\f5isapi.dll** を使用して、ファイル **f5isapi.dll** をブラウズし、[**OK**] をクリックします。
   c) [**Set Extension Status to Allowed**] ボックスをチェックし、[**OK**] をクリックします。
   エクステンションリストで、値 **F5 ISAPI** が [**Allowed**] として表示されます。

データ収集エージェント F5.IsHandler.dll のインストール

F5.IsHandler.dll ファイルは、IIS バージョン 6.0 と 7.0 にインストールすることができます。

データ収集エージェント F5.IsHandler.dll を IIS 6.0 サーバにインストールするには

1. **C:\Inetpub** の下に **scripts** ディレクトリを作成します（**C:\Inetpub\scripts**）。
2. **scripts** の下に **\bin** ディレクトリを作成します (**C:\Inetpub\scripts\bin**)。
3. **scripts** ディレクトリのプロパティを、**scripts and executables**（スクリプトと実行可能ファイル）に設定します。
4. **F5.IsHandler.dll** ファイルを **C:\Inetpub\scripts\bin** ディレクトリにコピーします。

5. **C:\Inetpub\scripts** ディレクトリ内に **web.config** ファイルを作成します。たとえば、図 B.1 のようなファイルが作られることになります。

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <system.web>
        <httpHandlers>
            <clear />
            <add verb="*" path="F5Isapi.dll"
type="F5.IsHandler" />
        </httpHandlers>
    </system.web>
</configuration>
```

***図 B.1** バージョン 6.0 を実行する IIS サーバで作成される **web.config** ファイルの例*

6. [Start] メニューから [Control Panel] を選択して、[**Administration Tools**] をダブルクリックします。
7. [**Internet Information Services**] をダブルクリックすると、IIS 管理コンソールが開きます。
8. ローカルコンピュータの名前を選択します。
9. **ASP.NET v2.0<build_number>** ファイルを有効にします。
   a) [**Web Server Extensions**] を選択します。
   b) [**ASP.NET v2.0<build_number>]** を選択します。
   c) [**Allow**] をクリックします。
10. 新しい仮想ディレクトリ **scripts** を作成します。
    a) [**Websites**] → [**<Default Web Site>]** を選択します。
    b) [**<Default Web Site>**] を右クリックして、[New] → [Virtual Directory] の順に選択します。
    c) [**Next**] をクリックします。
    d) エイリアスを **scripts** と入力して、[**Next**] をクリックします。
    e) ステップ 1 で作成したディレクトリ（**C:\Inetpub\scripts**）を入力して、[**Next**] をクリックします。
    f) 再度 [**Next**] をクリックします。
    g) [**Finished**] をクリックします。
11. **F5.IsHandler.dll** ファイルのアプリケーションプールを作成します。
    a) [**Application Pools**] を右クリックして、[New] → [Application Pool] の順に選択します。

b) [**Application Pool ID**] ボックスに「 **F5 Application Pool**」と入力して、[**OK**] をクリックします。
これで、**scripts** 仮想ディレクトリに対してアプリケーションプール、マッピング、ディレクトリセキュリティ、ASP.NET を設定するための手順に進むことができます。

12. **scripts** を右クリックして、[**properties**] を選択します。
13. アプリケーションプールを設定します。
    a) [Virtual Directory] タブをクリックします。
    b) [**Application Pool**] リストから [**F5 Application Pool**] を選択します。
14. マッピングを設定します。
    a) [**Configuration**] ボタンをクリックします。
    b) Application Configuration 画面の [Mappings] タブで、[**Add**] をクリックします。
    c) 実行ファイルとして、ファイル名 **C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727\aspnet_isapi.dll** を入力します。
    d) ファイル名の拡張子として **.dll** を入力します。
    e) [**Check that file exists**] ボックスをオフにして、[**OK**] をクリックします。
    f) Application Configuration 画面で [**OK**] をクリックします。
15. ディレクトリセキュリティを設定します。
    a) [Directory Security] タブをクリックします。
    b) [**Edit**] ボタンをクリックします。
    c) [**Anonymous Access and Integrated Windows**] ボックスをクリックして、認証を無効にします。
    d) [**Basic Authentication**] ボックスをオンにして、[**OK**] をクリックします。

    ***注**：ローカル認証を行わない場合は、デフォルトのドメインや領域を設定する必要はありません。*
16. ASP.NET プログラムを設定します。
    a) [ASP.NET] タブをクリックします。
    b) ASP.NET バージョンリストから、**2.0.<buildnumber>**（例：**2.0.50727**）を選択します。
17. **scripts** の [Properties] ページで、[**OK**] をクリックします。
18. IIS メタベースへのアクセスを設定します。
    a) **aspnet_regiis –ga <ASP.NETUsername>** コマンドを実行します。
    b) **http://support.microsoft.com/?kbid=267904** を参照してください。

データ収集エージェント F5.IsHandler.dll を IIS 7.0 サーバにインストールするには

1. **C:\Inetpub** の下に **scripts** ディレクトリを作成します（**C:\Inetpub\scripts**）。
2. **scripts** の下に **\bin** ディレクトリを作成します（**C:\Inetpub\scripts\bin**）。
3. **F5.IsHandler.dll** ファイルを **C:\Inetpub\scripts\bin** ディレクトリにコピーします。
4. **C:\Inetpub\scripts** ディレクトリ内に **web.config** ファイルを作成します。
   たとえば、図B.2のようなファイルが作られることになります。

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <system.webServer>
        <handlers>
            <clear />
            <add name="F5IsHandler" path="F5Isapi.dll"
verb="*" type="F5.IsHandler" modules="ManagedPipelineHandler"
scriptProcessor="" resourceType="Unspecified"
requireAccess="Script" preCondition="" />
        </handlers>
        <security>
            <îFèÿ>
                <anonymousAuthentication enabled="false" />
            </authentication>
        </security>
    </system.webServer>
</configuration>
```

***図 B.2** バージョン 7.0 を実行する IIS サーバで作成される **web.config** ファイルの例*

5. **appcmd** コマンドを使って、マシンレベルの **applicationHost.config** ファイルでオーバーライドモードを設定し、匿名認証のオーバーライドを許可します。コマンドの例:

```
appcmd set config "Default Web Site/scripts"
/section:anonymousAuthentication /overrideMode:Allow
/commit:APPHOST
```

6. **F5.IsHandler.dll** ファイルの新しいアプリケーションプールを設定します。
   a) [Start] メニューから、[Control Panel] を選択します。
   b) [Administrative Tools] を選択します。
   c) [Internet Information Services (IIS) Manager] を選択します。
   d) [**Connections**] から、[**<MachineName> (MachineName\UserName)**] を選択します。
   e) [Application Pools] メニューを右クリックして、[Add Application Pool] を選択します。

f) [**Name**] ボックスに、**F5 Application Pool** と入力します。

g) [**OK**] をクリックします。

7. 新しいアプリケーション「**scripts**」を作成します。

a) [**Web Sites**] → [**<MachineName>**] を選択します。

b) [**<MachineName>**] を右クリックして、[**Add Application**] を選択します。

c) [**Alias**] ボックスに「**scripts**」と入力します。

d) アプリケーションプールを変更するには、[**Select**] をクリックします。

e) 物理パスとして、ステップ 1 で作成したディレクトリ **(C:\Inetpub\scripts\)** を入力します。

f) [**OK**] をクリックします。

8. [**Authentication**] 設定を [**Basic Authentication**] に変更します。

a) **scripts** を選択します。

b) センターペインで、[**Authentication**] をダブルクリックします。

c)「**Basic Authentication**」を除く [**Authentication**] 下のすべてのアイテムのステータスが **Disabled** になっていることを確認します。
認証アイテムを有効化 / 無効化するには、アイテムの名前を右クリックして、Enable か Disable のどちらかを選択します。

プラグインをインストールしたら、WMI モニタを設定して、設定したモニタをプールメンバに関連付け、ロードバランシング方式として Dynamic Ratio を設定する必要があります。

- WMI モニタの設定については、第 12 章「*モニタの設定*」を参照してください。
- カスタムモニタのプールメンバへの関連付けについては、第 4 章「*ロードバランシングプールの設定*」を参照してください。
- プールのロードバランシング方式を Dynamic Ratio 方式に設定する方法については、第 4 章「*ロードバランシングプールの設定*」を参照してください。

SNMP DCA および SNMP DCA Base モニタの実装

BIG-IP には、SNMP データ収集エージェントが装備され、UC Davis エージェントや Windows 2000 Server エージェントなど、多様なタイプのリモート SNMP エージェントをクエリできます。

BIG-IP には 2 つのモニタタイプがあり、SNMP エージェントを使用するサーバにパフォーマンスモニタを作成するときに使用できます。次の 2 つのモニタタイプがあります。

- **SNMP DCA**
  このモニタは、CPU、メモリ、ディスクメトリックにデフォルト値を使用する場合、または新しい値を指定する場合に使用します。このテンプレートを使用する場合は、収集する他のタイプのメトリックの値も指定できます。
- **SNMP DCA Base**
  このモニタは、CPU、メモリ、およびディスク使用状況以外のメトリックにデフォルト値を使用する場合、または値を指定する場合に使用します。このモニタを使用すると、CPU、メモリ、およびディスクメトリックの値は省略されます。

サーバで SNMP エージェントを使用して Dynamic Ratio ロードバランシングを設定するには、次の 3 つのタスクを実行します。SNMP DCA または SNMP DCA Base モニタの設定、該当するプールメンバへのモニタの関連付け、プールでのロードバランシング方式の Dynamic Ratio 方式への設定です。詳細については、このガイドの次の章または項を参照してください。

- SNMP DCA および SNMP DCA Base モニタの設定については、第 12 章「*モニタの設定*」を参照してください。
- モニタのプールへの関連付けについては、第 4 章「*ロードバランシングプールの設定*」を参照してください。
- プールのロードバランシング方式を Dynamic Ratio 方式に設定する方法については、第 4 章「*ロードバランシングプールの設定*」を参照してください。

MSSQL モニタの実装

MSSQL タイプのモニタを使用するには、JDBC Java™ Archive（JAR）ファイルを Microsoft Web サイトからダウンロードして、BIG-IP にインストールしておく必要があります。

Microsoft JDBC ファイルをダウンロードしてインストールするには

1. インターネットブラウザから、**www.microsoft.com** に移動します。
2. このWebサイトから、**ダウンロードセンター**にアクセスします。
3. ページの上端付近にある [**Search**] リストで、[**All Downloads**] が選択されていることを確認します。
4. [**Search**] リストの右側のテキストフィールドに、**JDBC Driver** と入力します。
5. [**Go**] をクリックします。
   有効なドライバの一覧が表示されます。
6. 実行しているサービスパックに応じて、[**Microsoft SQL Server 2000 Driver for JDBC**] リンクの中からいずれか 1 つをクリックします。

   注: BIG-IP では、Microsoft SQL Server 2005 をサポートしていません。
7. [**Files in This Download**] というセクションまでページをスクロールします。
8. ファイルリストで、**mssqlserver.tar** というファイルを探します。
9. 画面右側の [**Download**] をクリックします。

   *注: Windows パッケージではなく、UNIX **.tar** ファイルをダウンロードしてください。正しくない場合は、前の画面に戻ります。*
10. 必要に応じて [**Save**] をクリックすれば、ファイルがディスクに保存されます。
11. Linux ディレクトリを作成して、**.tar** ファイルをそこに移動します。
12. 以下のコマンドを使ってファイルを展開します。

    ```
    tar xvf mssqlserver.tar
    ```

    *注: このコマンドを実行すると、**EULA.txt**、**install.ksh**、**msjdbc.tar**、**read.me** の 4 つのファイルが展開されます。*
13. 展開されたファイルを以下の手順でインストールします。

    a) コマンドラインプロンプトで「**install.ksh**」と入力します。

    b) 決められた指示に従います。
    これによって **msjdbc.tar** ファイルが解凍され、いくつかのサブディレクトリが作成されます。

    c) **lib** サブディレクトリを探します。
    このディレクトリには、3 つの JAR ファイルが含まれています。

d) 3 つの JAR ファイルを BIG-IP ディレクトリ **/usr/bin/monitors/builtins/** にコピーします。

14. 先に作成した Linux ディレクトリを再帰的に削除します。この手順はオプションです。

JAR ファイルをインストールした後、BIG-IP を再起動するか、または次のコマンドを実行することをお勧めします。

```
/usr/bin/monitors/builtins/DB_monitor cmd quit
```

システムが再起動されることで、BIG-IP は次に MSSQL モニタを実行したときに、新しくインストールした JAR ファイルを認識します。

C

OpenSSL ユーティリティの使い方

- BIG-IP の OpenSSL について
- クライアントシステム証明書の作成
- Web サイト証明書の作成
- 証明書失効の使い方
- その他の証明書関連タスクの実行

BIG-IP の OpenSSL について

SSLトラフィックの終端/開始にBIG-IPを使用している場合、OpenSSLユーティリティを使って以下のことを行うことができます。

- クライアントシステム証明書の作成
- Web サイト証明書の作成
- 証明書失効リスト（CRL）の作成
- その他の証明書関連タスクの実行（証明書の確認や PEM 形式への変換など）

クライアントシステム証明書の作成

クライアントと BIG-IP 間でクライアント側の認証を行うために、そのクライアントの証明書を作成することができます。

クライアント証明書を作成するには

1. BIG-IP のプロンプトにアクセスします。
2. クライアントキーを作成します。たとえば、次のようになります。

   ```
   openssl genrsa -rand .rand -out auser1.key 1024
   ```

3. 上記で作成したキーを使って、クライアント証明書要求を作成します。たとえば、次のようになります。

   ```
   openssl req -new -out auser1.req -key auser1.key
   ```

4. クライアント証明書（LDAP CRL 分散ポイント有 / 無）を作成します。なお、dirname ベースのアドレスである配布点が組み込まれた証明書を作成するには、**OpenSSL 0.9.8.x** 以上のバージョンを使用する必要があります(**dirname** はファイル名の末尾部分を取り除くユーティリティであり、結果、そのファイルが含まれるディレクトリのパス名が残ることになります)。

   以下の例では、**auser1.crt** という証明書を作成しています。

   - LDAP CRL 分散ポイントを含むクライアント証明書を作成するには、**openssl x509** コマンドを以下のように使用します。

     ```
     openssl x509 -req -in auser1.req -out auser1.crt \
     -CAkey bigmirror-ca.key -CA bigmirror-ca.crt \
     -days 300 -CAcreateserial -CAserial serial \
     -extensions crl_ext -extfile bigmirror-ca.ext
     ```

   - LDAP CRL 分散ポイントを含まないクライアント証明書を作成するには、**openssl x509** コマンドを以下のように使用します。

     ```
     openssl x509 -req -in auser1.req -out auser1.crt \
     -CAkey bigmirror-ca.key -CA bigmirror-ca.crt \
     -days 300 -CAcreateserial -CAserial serial
     ```

5. 上記で作成したキーと証明書ペアを使って、PKCS12 ファイルを作成します。
   たとえば、次のようになります。

   ```
   openssl pkcs12 -export -in auser1.crt -inkey \
   auser1.key -out auser1.p12 -name "auser1 pkcs12"
   ```

Web サイト証明書の作成

Web サイトと BIG-IP 間でサーバ側の認証を行うために、その Web サイトの証明書を作成することができます。

Web サイト証明書を作成するには

1. BIG-IP のプロンプトにアクセスします。
2. キーを作成します。たとえば、次のようになります。

   ```
   openssl genrsa -rand .rand -out www.test.net.key 1024
   ```

3. ステップ 1 で作成したキーを使って、証明書要求を作成する。以下のように作成することができます。

   ```
   openssl req -new -key www.test.net.key -out \
   www.test.net.req
   ```

4. ステップ 2 で作成した証明書要求を使って、その Web サイトの名前のついた証明書を作成します。
   - LDAP CRL 分散ポイントを含む証明書を作成するには、**openssl x509** コマンドを以下のように使用します。

     ```
     openssl x509 -req -in www.test.net.req -out \
     www.test.net.crt -CAkey bigmirror-ca.key -CA \
     bigmirror-ca.crt -days 300 -CAcreateserial \
     -CAserial serial -extensions crl_ext \
     -extfile bigmirror-ca.ext
     ```

   - LDAP CRL 分散ポイントを含まない証明書を作成するには、**openssl x509** コマンドを以下のように使用します。

     ```
     openssl x509 -req -in www.test.net.req \
     -out www.test.net.crt -CAkey bigmirror-ca.key -CA
     bigmirror-ca.crt -days 300 -CAcreateserial \
     -CAserial serial
     ```

証明書失効の使い方

[**OpenSSL**] ユーティリティを使って、CRL（証明書失効リスト）を作成することができます。BIG-IP は、CRL をチェックして、認証用に提示されたクライアントまたはサーバの証明書が失効しているかどうかを確認します。

OpenSSL ユーティリティを使って証明書を失効させることも可能です。

証明書失効リストを作成するには

1. BIG-IP システムプロンプトから、シリアルまたはインデックスオプションの設定ファイルを作成します。
   たとえば、次のようになります。

   ```
   echo -e \
   'default_ca=ca\n[ca]\ndatabase=index.txt\nserial=serial'
   > bigmirror-ca.config
   ```

2. BIG-IP システムプロンプトから、有効期限が 30 日の CRL を作成します。たとえば、次のようになります。

   ```
   openssl ca -config bigmirror-ca.config -gencrl -crldays \
   30 -keyfile bigmirror-ca.key -cert bigmirror-ca.crt \
   -out bigmirror-ca.crl
   ```

証明書を無効にするには

BIG-IP システムプロンプトから、**openssl** コマンドを使って、クライアント証明書を無効にします。たとえば、以下の例ではクライアント証明書 **auser1.crt** を無効にしています。

```
openssl ca -config bigmirror-ca.config -keyfile \
bigmirror-ca.key -cert bigmirror-ca.crt -revoke auser1.crt
```

◆注

CRLDP 認証モジュールを使用する場合は、CRL がリモートの LDAP データベースに ASN.1 DER (Abstract Syntax Notation.1 Distinguished Encoding Rules) 形式で保存されている必要があります。

その他の証明書関連タスクの実行

このほかにも多数の SSL 証明書関連タスクを、**OpenSSL** ユーティリティを使って実行することができます。同ユーティリティには BIG-IP システムプロンプトからアクセスできるようになっています。

証明書を検証するには

証明書を検証するには、以下のコマンドを使用します。

```
openssl verify -CAfile bigmirror-ca.crt www.test.net.crt
```

CRL を確認するには

CRL を確認するには、以下のコマンドを使用します。

```
openssl crl -in bigmirror-ca.crl -text -noout
```

証明書情報を確認するには

証明書情報を確認するには、以下のコマンドを使用します。

```
openssl x509 -in www.test.net.crt -text -noout
```

証明書を PEM 形式に変換するには

証明書を PKCS12（P12 or.PFX）形式から PEM 形式に変換するには、以下のコマンドを使用します。

```
openssl pkcs12 -in auser1.p12 -out auser1.pem
```

RSA キーにパスワードを追加するには

RSA キーにパスワードを追加するには、以下のコマンドを使用します。

```
openssl rsa -in auser1.key -out auser1-enc.key -des3 \
-passout pass:secret
```

RSA キーからパスワードを削除するには

RSA キーからパスワードを削除するには、以下のコマンドを使用します。

```
openssl rsa -in auser1-enc.key -out auser1.key \
-passin pass:secret
```

用語集

ARP（Address Resolution Protocol：アドレス解決プロトコル）

ARP は業界標準のプロトコルで、ホストの MAC（Media Access Control）アドレスをその IP アドレスに基づいて判定します。

bigtop

bigtop ユーティリティは、統計的監視ユーティリティで、BIG-IP に付属しています。このユーティリティによって、リアルタイムの統計情報が得られます。

BIND（Berkely Internet Name Domain）

BIND は、DNS（Domain Name System）の最も一般的な実装です。BIND によって、BIG-IP は IP アドレスに一致するドメイン名を取得できます。詳細については、**http://www.isc.org/products/BIND** を参照してください。

BPDU（bridge protocol data unit）

BPDU は、スパニングツリープロトコルがレイヤ 2 デバイス間に送信する特殊なパケットで、冗長パスを判定し、ループを解決します。「*STP（Spanning Tree Protocol）*」、「*RSTP（Rapid Spanning Tree Protocol）*」、「*MSTP（Multiple Spanning Tree Protocol）*」も参照してください。

BYE トランザクション

BYE トランザクションとは、2 つのアプリケーション間の接続を閉じる準備が整った時点で、アプリケーションがもう一方のアプリケーションに送信するメッセージです。

Call-ID

Call-ID は、アプリケーション間でやり取りされる一連のメッセージを 1 つのグループにまとめるグローバル一意識別子です。

Certificate Revocation List Distribution Point (CRLDP)

「*CRLDP（Certificate Revocation List Distribution Point）*」を参照してください。

Cipher

Cipher は、コンピュータシステムが使用する暗号化 / 復号化アルゴリズムで、SSL プロトコルを使用したデータ転送に使用されます。

CMP（クラスタリングマルチプロセッシング）

CMP は、内部パフォーマンスを向上させるための機能で、特定のマルチプロセッサの BIG-IP で使用できます。BIG-IP では、CMP を使用することにより、アプリケーショントラフィックを同時に処理するための TMM サービスのインスタンスが複数作成されます。

Cookie パーシステンス

Cookie パーシステンスは、パーシステンスのモードの 1 つで、BIG-IP がパーシステンス接続情報を Cookie に保存できます。

CRL（証明書失効リスト）

CRL は、認証するシステムが、要求側のシステムが認証用に提示した SSL 証明書が無効になっていないか確認するためのリストです。

CRLDP（Certificate Revocation List Distribution Point）

CRLDPは業界標準のプロトコルで、ネットワーク上のデバイスのSSL証明書失効を管理します。

CRLDP 認証モジュール

CRLDP 認証モジュールは、ユーザ作成のモジュールで、BIG-IP 上で実装してCRLDPプロトコルを使用してクライアントトラフィックを認証します。ネットワーク上のクライアント SSL 証明書の無効状態の管理を目的とします。

Dynamic Ratio ロードバランシング方式

Dynamic Ratio モードは、Ratio モード（「*Ratio 方式*」を参照）とよく似ていますが、比率の重み付けがサーバの連続的な監視に基づいているため、常に変化します。Dynamic Ratio ロードバランシングは、RealNetworks® RealServer プラットフォーム、WMI（Windows Management Instrumentation）を搭載した Microsoft® Windows® プラットフォーム、および UC Davis SNMP エージェントまたは Windows 2000 Server SNMP エージェントを搭載したサーバで実装できます。

EAV（Extended Application Verification）

EAV は、ノード上のアプリケーションをリモートで実行して、アプリケーションを検証するヘルスチェックです。EAV ヘルスチェックは、BIG-IP で使用できる 3 つのタイプのヘルスチェックの 1 つです。「*ヘルスチェック*」、「*ヘルスモニタ*」、「*外部モニタ*」も参照してください。

ECV（Extended Content Verification）

ECV は、ノードが返す特定のコンテンツに基づき、ノードが **up** か **down** かを判定するヘルスチェックです。ECV ヘルスチェックは、BIG-IP で使用できる 3 つのタイプのヘルスチェックの 1 つです。「*ヘルスチェック*」も参照してください。

Fastest 方式

Fastest モードはロードバランシング方式で、現在アクティブなすべてのノードのうち最も応答が速いものに基づいて新しい接続を渡します。

FDDI（Fiber Distributed Data Interface）

FDDI はマルチモードプロトコルで、光ファイバー上で最大 100Mbps の速度でデータを転送します。

HTTP チャンク

HTTP チャンクは、チャンクエンコードとして知られる HTTP/1.1 の機能で、HTTP メッセージを複数の部分に分解できます。チャンクは通常、応答を送信する際にサーバで使用されます。

HTTP 変換

BIG-IP が HTTP 変換を実行すると、システムはサーバ側 HTTP リクエストの **Connection** ヘッダを操作して、接続を開いたままにします。「*接続パーシステンス*」も参照してください。

HTTP リダイレクト

HTTP リダイレクトは、クライアントに「HTTP 302 Object Found」メッセージを送信します。プールに HTTP リダイレクトを設定すると、プールメンバが **down** の場合に、別のノードまたはバーチャルサーバにクライアントを送信できます。

ICMP（Internet Control Message Protocol）

ICMP は、宛先アドレスへのルート情報の判定に使用される、インターネット通信プロトコルです。

IPsec

IPsec（Internet Security Protocol）は、インターネットのネットワーク層にセキュリティを提供する通信プロトコルで、このプロトコル上で実行するアプリケーションに対する必要条件はありません。

iRule

iRule はユーザ作成のスクリプトで、BIG-IP を通過する接続の動作を制御します。iRules™ は F5 ネットワークスの機能で、特定の接続をデフォルト以外のロードバランシングプールにダイレクトする場合によく使用されます。ただし iRule では、安全なネットワークアドレス変換の実装やセッションパーシステンスの実現など、その他のタスクも実行できます。

i モード

i モード ® は、NTT DoCoMo, Inc. が作成したサービスで、携帯電話ユーザがインターネットにアクセスできます。

JAR ファイル

JAR ファイルは、Java™ Archive（JAR）ファイル形式のファイルで、複数のファイルを 1 つのアーカイブファイルにバンドルできます。通常、JAR ファイルにはクラスファイルと、アプレットおよびアプリケーションに関連する補助的なリソースが含まれています。

JDBC

JDBC は、Java™ テクノロジです。様々な SQL データベースに対してのデータベース管理システム（ DBMS）接続性と 、スプレッド シートやフラット ファイルなどその他の表形式のデータソースへのアクセスを提供する、アプリ ケーション プログラミング インターフェイスです。

Kerberos プロトコル

Kerberos プロトコルはネットワーク認証プロトコルで、安全でないネットワークを介して通信を行う個人が、安全な方法でその身元を他方に証明できるようにします。主にクライアント - サーバモデルでの使用を目的としていて、相互認証により、ユーザとサーバの両方がそれぞれの身元を確認します。

Kilobytes/Second モード

Kilobytes/Second モードは、現在処理しているキロバイト毎秒が最も少ないサーバに接続を分配する、ダイナミック ロードバランシング モードです。

LACP（Link Aggregation Control Protocol）

LACP はトランク内のリンクを集約する業界標準のプロトコルで、帯域幅を増加させてリンクフェイルオーバーを提供します。

LDAP（Lightweight Directory Access Protocol）

LDAP は、電子メールプログラムがサーバから宛先情報を検索する際に使用するインターネットプロトコルです。

LDAP クライアント証明書 SSL 認証モジュール

LDAP クライアント証明書 SSL 認証モジュールは、BIG-IP 上に実装するユーザ作成のモジュールで、SSL クライアント資格情報とリモート LDAP サーバを使用してクライアントトラフィックを認証します。

LDAP 認証モジュール

LDAP 認証モジュールは、BIG-IP 上に実装するユーザ作成のモジュールで、リモート LDAP サーバを使用してクライアントトラフィックを認証します。

Least Connections 方式

Least Connections モードは、現在処理しているオープン接続が最も少ないサーバに接続を分配する、ダイナミック ロードバランシング方式です。

LACP（Link Aggregation Control Protocol）

「*LACP（Link Aggregation Control Protocol）*」を参照してください。

MAC（Media Acces Control）

MAC は、ワークステーションが転送メディアへのアクセス権を取得する方法を定義するプロトコルで、LAN への参照に一般的に使用されています。IEEE LAN では、MAC 層はデータリンク層プロトコルの下位サブレイヤです。

MAC アドレス

MAC アドレスは、Ethernet ネットワーク上のハードウェアデバイスを識別するのに使用されます。

MCPD サービス

MCPD（Master Control Program Daemon）サービスは、BIG-IP の設定データを管理します。

MSRDP パーシステンス

MSRDP パーシステンスは、Microsoft® Remote Desktop Protocol（RDP）サービスを実行しているクライアントとサーバ間のセッションを追跡します。

MSTP（Multiple Spanning Tree Protocol）

MSTP は IEEE で規定された、高度なスパニングツリープロトコルです。STP や RSTP とは異なり、MSTP は VLAN を認識します。そのため MSTP リージョンの概念が組み込まれています。「*STP（Spanning Tree Protocol*」、「*RSTP（Rapid Spanning Tree Protocol）*」も参照してください。

MSTP リージョン

MSTP リージョンは、特定の設定項目に同じ値を持つレイヤ 2 デバイスのグループです。デバイスがリージョンを構成すると、冗長パスのブロックまたはアンブロックの際にスパニングツリーアルゴリズムでは VLAN を対象に含めます。

NAT（Network Address Translation：ネットワークアドレス変換）

NAT は、BIG-IP が管理する特定のノードを特定する、外部ネットワークに対するエイリアス IP アドレスです。

NT LAN マネージャ

*NTLM（NT LAN マネージャ）*をご覧ください。

NTLM（NT LAN マネージャ）

NTLM は、ユーザのパスワードをネットワークで送信することなく、暗号化された要求/応答プロトコルを使ってユーザの認証を行う業界標準テクノロジです。

Observed 方式

Observed 方式は、現在処理している接続が最も少ないサーバに接続を分配する、ダイナミックロードバランシング方式です。Least Connections 方式と違い、Observed 方式では接続数を継続的に追跡し、ロードバランシングの比率を決定します。

OCSP（Online Certificate Status Protocol）

OCSP は、認証システムがデジタル署名された SSL 証明書の無効状態のチェックに使用するプロトコルです。OCSP は、CRL（証明書失効リスト）の代替手段として使用できます。
「*CRL（証明書失効リスト）*」も参照してください。

OCSP 認証モジュール

OCSP 認証モジュールは、BIG-IP 上に実装するユーザ作成のモジュールで、リモート OCSP レスポンダを使用してクライアントトラフィックを認証します。OCSP 認証モジュールの目的は、クライアント SSL 証明書の無効状態をチェックすることです。

OCSP レスポンダ

OCSP レスポンダは、BIG-IP などの認証サーバに SSL 証明書失効状態を伝える場合に使用される、外部サーバです。

OCSP レスポンダオブジェクト

レスポンダオブジェクトは、BIG-IP 上のソフトウェアアプリケーションで、OCSP レスポンダと通信して、クライアントまたはサーバ SSL 証明書の失効状態をチェックします。

OneConnect™

OneConnect 機能は、サーバ側接続を開いた状態に保ち再利用のためにプールすることで、ネットワーク接続の利用を最適化します。

PAM（Pluggable Authentication Module）

PAM モジュールは、サーバアプリケーションがクライアントトラフィックの認証に使用するソフトウェアモジュールです。PAM モジュールはモジュラー設計されているため、サーバアプリケーションへの認証メカニズムの追加、置換、削除を、アプリケーションに与える影響を最小限にして実行できます。PAM モジュールの例として、リモート LDAP（Lightweight Directory Access Protocol）サーバを使用してクライアントトラフィックを認証するアプリケーションがあります。「*LDAP（Lightweight Directory Access Protocol）*」も参照してください。

QoS（Quality of Service）レベル

Quality of Service（QoS）レベルは、ネットワーク機器が識別子に基づいてトラフィックを個別に特定して処理するための手段です。基本的に、パケットで指定された QoS レベルによって、そのパケットのスループットポリシーが実施されます。

RADIUS（Remote Authentication Dial-in Service）

RADIUSは､リモート ユーザ認証とアカウンティングを実行するサーバです。主にインターネット サービスプロバイダが利用しますが、ワークステーションで一元化した認証および（または）アカウンティングサービスを必要とするあらゆるネットワークで使用することもできます。

RADIUS 認証モジュール

RADIUS 認証モジュールは、ユーザ作成のモジュールで、BIG-IP 上に実装し、リモート RADIUS サーバを使用してクライアントトラフィックを認証します。

RAM キャッシュ

RAM キャッシュは、BIG-IP の RAM に保存される HTTP オブジェクトのキャッシュで、以降の接続で再利用され、バックエンドサーバの負荷を低減できます。

ratio

ロードバランシング用にバーチャルサーバに重みを割り当てるパラメータです。

Ratio 方式

Ratio ロードバランシング方式は、各バーチャルサーバに割り当てられた重み付けに比例して､バーチャルサーバのアレイに接続を分散させます。

Real Time Streaming Protocol（RTSP）

「*RTSP（Real Time Streaming Protocol）*」を参照してください。

RFC 1918 アドレス

RFC 1918 アドレスは、IETF RFC 1918 で規定された、プライベートクラスの範囲にある IP アドレスのことです。

Round Robin モード

Round Robin モードは、指定されたサーバの順序で接続を分散するスタティックロードバランシングモードです。Round Robin モードでは、順序内で次に使用できるサーバに接続要求を送信します。

RSTP（Rapid Spanning Tree Protocol）

RSTP は IEEE で規定された、STP（スパニングツリープロトコル）の拡張バージョンです。RSTP は、STP よりも高速なスパニングツリーパフォーマンスを実現します。「*STP（Spanning Tree Protocol）*」および「*MSTP（Multiple Spanning Tree Protocol）*」も参照してください。

RTSP（Real Time Streaming Protocol）

RTSP は、オーディオやビデオのような連続的メディアの複数の同期ストリームを確立、制御します。

SCTP（Stream Control Transmission Protocol）

SCTP は、シグナリングデータを転送するメッセージ指向型のアプリケーション用に設計された、汎用の業界標準転送プロトコルです。

Secure Via ヘッダ

Secure Via ヘッダは、BIG-IP により SIP メッセージヘッダに挿入されるデータであり、SIP メッセージの送信元を示します。「*Via ヘッダ*」と「ヘッダ」も参照してください。

Session Initiation Protocol（SIP）

「*SIP（Session Initiation Protocol）*」をご覧ください。

SIP（Session Initiation Protocol）

SIP は、複数の参加者で構成されるセッションを管理するアプリケーションレイヤプロトコルであるため、リアルタイムのメッセージ、音声、データ、および映像が有効化されます。

SIP パーシステンス

SIP パーシステンスは、UDP 経由で SIP（Session Initiation Protocol）メッセージを受信するサーバが使用するパーシステンスです。SIP はリアルタイムのメッセージ、音声、データ、および映像を有効にするプロトコルです。

SNAT（セキュアネットワークアドレス変換）

SNAT は、BIG-IP で設定できる機能です。SNAT は、外部ネットワーク上のホストに接続する場合に、1 つ以上のノードが送信元 IP アドレスとして使用できる、転送可能なエイリアス IP アドレスを定義します。「*標準 SNAT*」と「*インテリジェント SNAT*」も参照してください。

SNAT プール

SNAT プールは、1 つ以上の元の IP アドレスにマッピングできる変換アドレスのプールです。SNAT プールの変換アドレスは、セルフ IP アドレスではありません。

SNMP（Simple Network Management Protocol）

SNMP はインターネット標準のプロトコルで、STD15 RFC 1157 で定義され、IP ネットワーク上のノード管理のために開発されました。

SSH

SSH はセキュアリモートログインと、非セキュアネットワーク上の他のセキュアネットワークサービス用のプロトコルです。

SSL（Secure Sockets Layer）

SSL はネットワーク通信プロトコルで、データを安全に転送する方法として公開鍵テクノロジを使用しています。

SSL パーシステンス

SSL パーシステンスは、SSL セッション ID を使用して、終了していない SSL セッションを追跡するタイプのパーシステンスです。

SSL プロファイル

SSL プロファイルは設定ツールで、クライアントおよびサーバから SSL 接続を終了および開始するときに使用します。

STP（Spanning Tree Protocol）

STP は IEEE で定義されたプロトコルで、1 つ以上の外部スイッチが BIG-IP に並列に接続されている場合に、設定でのループを解決します。「*RSTP（Rapid Spanning Tree Protocol）*」と「*MSTP（Multiple Spanning Tree Protocol）*」も参照してください。

Stream Control Transmission Protocol（SCTP）

「*SCTP (Stream Control Transmission Protocol)*」をご覧ください。

TACACS（Terminal Access Controller Access Control System）

TACACS は UNIX システムによく使用される、古いタイプの認証プロトコルです。TACACS では、リモートアクセスサーバはユーザのログインパスワードを認証サーバに転送できます。

TACACS+

TACACS+ は、それ以前の TACACS プロトコルを置き換えるために設計された認証メカニズムです。この 2 つのプロトコルに類似点はほとんどなく、したがって互換性はありません。

TACACS+ 認証モジュール

TACACS+ 認証モジュールは、ユーザ作成のモジュールで、BIG-IP 上に実装し、リモート TACACS+ サーバを使用してクライアントトラフィックを認証します。

Tcl

Tcl（Tools Command Language）は、業界標準のスクリプティング言語です。BIG-IP では、ユーザは Tcl を使用して iRules を記述します。

TMM（Traffic Management Microkernel）サービス

TMM サービスは、BIG-IP で実行されるプロセスで、製品のトラフィック管理の大半を実行します。

TMM スイッチインターフェイス

TMMスイッチインターフェイスは、BIG-IPがHTTPやSSLトラフィックのようなユーザアプリケーショントラフィックの転送に使用するインターフェイスです。このため、アプリケーショントラフィックのロードバランシングを実行する場合、BIG-IP は TMM スイッチインターフェイスを使用します。「*管理インターフェイス*」も参照してください。

TMM スイッチルート

TMM（Traffic Management Microkernel）スイッチルートは、TMM スイッチインターフェイスを経由してトラフィックを転送するルートです。管理インターフェイスは経由しません。

ToS（Type of Service）レベル

ToS レベルは、QoS（Quality of Service）に加えて、ネットワーク機器が識別子に基づいてトラフィックを個別に特定して処理するための手段です。

UIE（Universal Inspection Engine）

UIE は、ユニバーサルパーシステンスとユニバーサルコンテンツの切り替えを提供する機能で、ロードバランシング能力を向上させます。UIE には、式を構成するためのルール変数と関数があり、これはプール定義とルールで指定できます。

Via ヘッダ

Via ヘッダは、BIG-IP により SIP メッセージに挿入されるデータであり、SIP メッセージの送信元を示します。「*Secure Via ヘッダ*」と「ヘッダ」も参照してください。

VLAN（仮想ローカルエリアネットワーク）

VLAN は、ネットワークデバイスに接続されたインターフェイスを、論理的にグループ化したものです。VLAN を使用して、別のネットワークセグメントにあるデバイスを、論理的にグループ化できます。VLAN 内のデバイスは、レイヤ 2 ネットワーキングを使用して通信し、ブロードキャストドメインを定義します。

VLAN グループ

VLAN グループは、2 つ以上の VLAN を 1 つの VLAN グループにまとめたものです。VLAN グループの主な使用方法は、送信元と宛先ホストの両方が同じネットワーク上にある場合、トラフィックを正常に転送することです。

VLAN タグ

VLAN タグは IEEE 標準で、フレームのヘッダに挿入された識別番号で、宛先デバイスが属する VLAN を示します。VLAN タグは、1 つのインターフェイスが複数の VLAN にトラフィックを転送する場合に使用されます。

VLAN 名

VLAN 名は、VLAN を識別するために使用するシンボル名です。たとえば、**marketing** または **development** という名の VLAN を設定できます。「*VLAN（仮想ローカルエリアネットワーク）*」も参照してください。

WAP（Wireless Application Protocol）

WAP は、メーカー、ベンダー、およびテクノロジに依存せずに、インターネットアクセスや高度なテレフォニーサービスを実現するように設計された、ワイヤレスデバイスのためのアプリケーション環境および一連の通信プロトコルです。

WKS（well-known service）

WKS は、ポート **0** からポート **1023** 上のプロトコルで、特定のタイプのデータに広く使用されています。WKS（およびその対応するポート）の例としては、HTTP（ポート **80**）、HTTPS（ポート **443**）、FTP（ポート **20**）があります。

アーカイブ

アーカイブは、BIG-IP 設定データのバックアップコピーです。このアーカイブは、ユーザ設定セット（UCS）形式です。「*ユーザ設定セット（UCS）*」も参照してください。

アクティブモニタリング

*アクティブモニタリング*は、プールメンバやノードの状態を一定間隔で継続的にチェックします。「*パッシブモニタリング*」と「*シンプルモニタリング*」もあわせて参照してください。

アクティブユニット

アクティブユニットとは、冗長システムで現在接続のロードバランスを実行しているシステムです。冗長システムのアクティブユニットに障害が起きた場合に、スタンバイユニットに制御が渡り、接続のロードバランスを実行します。「*冗長システム*」も参照してください。

宛先アドレスのアフィニティパーシステンス

スティッキーパーシステンスとも呼ばれます。宛先アドレスアフィニティパーシステンスは、TCP および UDP プロトコルをサポートし、セッション要求をパケットの宛先 IP アドレスのみに基づいて同じサーバにダイレクトします。

アプリケーションセキュリティクラス

アプリケーションセキュリティクラスは HTTP Class プロファイルであり、Application Security Manager で設定するようになっています。

アプリケーションテンプレート

アプリケーションテンプレートは BIG-IP 設定ユーティリティ内のツールであり、アプリケーション固有のトラフィックを処理するためのシステムの設定を容易にします。アプリケーションテンプレートには Microsoft® SharePoint 用テンプレートなどがあります。

インターフェイス

BIG-IP の物理ポートは、インターフェイスと呼ばれます。

インテリジェント SNAT

インテリジェント SNAT は、SNAT プールを作成し、この SNAT プールを参照する iRule を作成することによって実装される SNAT です。「*SNAT*」も参照してください。

ウォッチドッグタイマーカード

ウォッチドッグタイマーカードは、BIG-IP のハードウェア障害を監視する、ハードウェアデバイスです。

親プロファイル

親プロファイルは、その値を他のプロファイルに伝播できるプロファイルです。親プロファイルは、デフォルトプロファイルまたはカスタムプロファイルのどちらでも可能です。「*プロファイル*」も参照してください。

外部 VLAN

外部 VLAN は、BIG-IP のデフォルト VLAN です。基本設定では、この VLAN の管理ポートはロックされています。通常の設定では、これは外部クライアントが内部サーバへの接続を要求する VLAN になります。

外部認証

外部認証は、リモートサーバを使用して BIG-IP へアクセスを試みるユーザまたはアプリケーションの認証を目的とするデータを保存するプロセスです。

外部モニタ

外部モニタは、ユーザが提供するヘルスモニタです 「*ヘルスチェック*」、「*ヘルスモニタ*」も参照してください。

カスタムプロファイル

カスタムプロファイルは、ユーザが作成するプロファイルです。カスタムプロファイルは、指定した親プロファイルのデフォルト設定を継承できます。「*親プロファイル*」と「*プロファイル*」も参照してください。

仮想アドレス

仮想アドレスは、BIG-IP で管理される 1 つ以上のバーチャルサーバに関連付けられた IP アドレスです。「*バーチャルサーバ*」も参照してください。

バーチャルサーバ

バーチャルサーバは、BIG-IP や他のホストサーバによって管理されているコンテンツサイトに関連付けられた仮想アドレスと仮想ポートの特定の組み合わせを指します。

仮想ポート

仮想ポートは、BIG-IP で管理される 1 つ以上のバーチャルサーバに関連付けられたポート番号またはサービス名です。仮想ポート番号は、クライアントプログラムが接続する先の TCP または UDP ポート番号と同じである必要があります。

管理インターフェイス

管理インターフェイスは、BIG-IP の特殊なポートで、管理トラフィックの管理に使用されます。**MGMT** と呼ばれ、管理インターフェイスは、ロードバランシングされるトラフィックなどのユーザアプリケーショントラフィックは転送しません。「*TMM スイッチインターフェイス*」も参照してください。

管理ルート

管理ルートは、特殊な管理（**MGMT**）インターフェイスを経由してトラフィックを転送するルートです。

クライアント側 SSL プロファイル

クライアント側 SSL プロファイルは、クライアントシステムから BIG-IP に流れる SSL トラフィックの動作を制御する SSL プロファイルです。

クラスタリングマルチプロセッシング（CMP）

「*CMP (クラスタリングマルチプロセッシング)*」を参照してください。

クローンプール

この機能は、プールが受信するすべてのトラフィックを複製し、そのトラフィックを複製プールに送信します。

ゲートウェイプール

ゲートウェイプールはルータのプールで、トラフィックを転送するために作成します。ゲートウェイプールを作成したら、TMM ルーティングテーブルエントリ内でプールをゲートウェイとして指定できます。

構成オブジェクト

構成オブジェクトはユーザが作成したオブジェクトで、BIG-IP はこれを使用して PAM 認証モジュールを実装します。作成する認証モジュールの各タイプに対して、それぞれ 1 つのタイプの構成オブジェクトがあります。「*PAM（Pluggable Authentication Module）*」も参照してください。

コンテンツの切り替え

コンテンツの切り替えによって、パケットに含まれるデータに基づいてプールを選択できます。

サーバ側 SSL プロファイル

サーバ側 SSL プロファイルは、BIG-IP と宛先サーバシステム間を流れる SSL トラフィックを制御する SSL プロファイルです。

サービス

TCP、UDP、HTTP、FTP などのサービスのことです。

サービスプロファイル

サービスプロファイルは、HTTP または FTP ネットワークトラフィックを管理する、BIG-IP のシステム設定ツールです。

最小アクティブメンバ

最小アクティブメンバは、プライオリティグループ内で、BIG-IP がそのグループに要求を送信するためにアクティブにする必要がある、メンバの数です。アクティブメンバの数がこの数より少ないと、要求は次にプライオリティの高いグループ（次にプライオリティ番号が小さいグループ）に送信されます。

最適化プロファイル

最適化プロファイルは、システムのインストール時に BIG-IP で作成されるカスタムプロファイルです。最適化プロファイルを使用することで、HTTP および TCP の 2 種類のトラフィックをもっとも効率的に処理することができます。

サブドメイン

サブドメインは、より高いレベルのドメインのサブセクションです。たとえば、**.com** は高レベルドメインで、**F5.com** は **.com** ドメインのサブドメインです。

参照リンク

参照リンクは、トランク内で番号が一番小さいインターフェイスで、リンクアグリゲーションに使用されます。

冗長システム

冗長システムは、フェイルオーバー用に設定された 2 台のユニットです。冗長システムでは 2 つのユニットがあり、1 つはアクティブユニットとして実行され、もう 1 つはスタンバイユニットとして実行されます。アクティブユニットに障害が発生すると、スタンバイユニットが処理を引き継いで接続要求を管理します。

承認

承認は、ログオンしたユーザに付与するシステムリソースへのアクセス権限のレベルを特定するプロセスです。

証明書

証明書は、信頼できる認証局が署名し、SSL ネットワークトラフィックに認証手段として使用される、オンライン認証情報です。

証明書失効リスト（CRL）

「*CRL（証明書失効リスト）*」を参照してください。

証明書の検証

証明書の検証は、SSL ハンドシェイクの一部で、クライアントの SSL 資格情報が信頼された認証局によって署名されていることを検証します。

シンプルパーシステンス

「*送信元アドレスのアフィニティパーシステンス*」を参照してください。

シンプルモニタリング

シンプルモニタリングは、ノードの状態が **up** であるか **down** であるかを判断します。シンプルモニタではプールメンバを監視することはなく（つまり、ノード上の個々のプロトコル、サービス、アプリケーションは監視の対象外）、ノード自体の監視のみが行われます。「*アクティブモニタリング*」と「*パッシブモニタリング*」もあわせて参照してください。

信頼できる CA ファイル

信頼できる CA ファイルは、認証のクライアントリクエストを処理する際に、認証システムが信頼できる認証局のリストを含むファイルです。信頼できる CA ファイルは、認証システム上に存在し、SSL ネットワークトラフィックの認証に使用されます。

スタティックセルフ IP アドレス

スタティックセルフ IP アドレスは、冗長システムの 2 つのユニット間で共有されない、セルフ IP アドレスです。

スタティックルート

スタティックルートは、レイヤ 3 デバイスのルーティングテーブルで明示的に設定する必要があるルートです。「*ダイナミックルート*」も参照してください。

スタンバイユニット

冗長システムのスタンバイユニットは、アクティブユニットで障害が発生したときにいつでもアクティブユニットに交代できるユニットです。

ステートミラーリング

ステートミラーリングは、接続とパーシステンス情報を冗長システムに保持する、BIG-IP の機能です。

スパニングツリー

スパニングツリーは、ネットワーク上のレイヤ 2 デバイスの論理的ツリー構造で、スパニングツリープロトコルアルゴリズムによって作成され、ネットワークループの解決に使用されます。

スパニングツリーインスタンス

スパニングツリーインスタンスは、スパニングツリープロトコルが作成する、指定された特定のスパニングツリーです。「*スパニングツリープロトコル*」も参照してください。

スパニングツリープロトコル

スパニングツリープロトコルは、IEEE 指定の、STP（Spanning Tree Protocol）、RSTP（Rapid Spanning Tree Protocol）、MSTP（Multiple Spanning Tree Protocol）として知られるプロトコルの集合です。BIG-IP では、これらすべてのプロトコルをサポートしています。「*STP（Spanning Tree Protocol）*」、「*RSTP（Rapid Spanning Tree Protocol）*」、「*MSTP（Multiple Spanning Tree Protocol）*」も参照してください。

セキュアネットワークアドレス変換（SNAT）

「*SNAT（セキュアネットワークアドレス変換）*」を参照してください。

セッションパーシステンス

同じクライアントから受信した一連の関連する接続で、同じセッション ID を持っています。パーシステンスを有効にすると、BIG-IP は同じセッション ID を持つすべての接続を同じノードに送信します。接続にロードバランシングは実行されません セッションパーシステンスと*接続パーシステンス*は、別のものです。

接続パーシステンス

接続パーシステンスは、ハンドシェイクを省くためにネットワーク接続を意図的に開いたままにする、最適化テクニックです。「*HTTP 変換*」と「*OneConnect™*」も参照してください。

コネクションプーリング

コネクションプーリングは、サーバ側接続を他のクライアントリクエストで再利用できるようにプールしておく、最適化機能です。コネクションプーリングによって、サーバ側のクライアントリクエストに対して開いておく必要のある新しい接続の数を減らすことができます。

設定済みモニタ

設定済みモニタは、BIG-IP が提供するヘルスモニタまたはパフォーマンスモニタです 設定済みモニタはそのまま使用できますが、変更したり削除したりすることはできません。「*モニタ*」も参照してください。

設定の同期

設定の同期は、BIG-IP の設定データを、冗長システムのピアユニットに複製するタスクです。

設定ユーティリティ

設定ユーティリティは、ブラウザベースのアプリケーションで、BIG-IP の設定に使用します。

セットアップユーティリティ

セットアップユーティリティを使用して、初期システム設定を処理できます。セットアップユーティリティは、設定ユーティリティのスタートページから実行できます。

セルフ IP アドレス

セルフ IP アドレスは、BIG-IP が所有する IP アドレスで、内部および外部 VLAN へのアクセスに使用できます。

送信元アドレスのアフィニティパーシステンス

シンプルパーシステンスとも呼ばれます。送信元アドレスのアフィニティパーシステンスは、TCP および UDP プロトコルをサポートし、パケットの送信元 IP アドレスのみに基づいて同じサーバにセッション要求をダイレクトします。

送信元処理

送信元処理とは、インターフェイスで受信パケットの送信元を書き換えることです。

ダイナミックルート

ダイナミックルートは、RIP のようなアドバンスドルーティングプロトコルがダイナミックにルーティングテーブルに追加するルートです。「*スタティックルート*」も参照してください。

タグ付きインターフェイス

タグ付きインターフェイスは、VLAN に割り当てられるインターフェイスで、BIG-IP はそのインターフェイスを通過したフレームのヘッダに VLAN タグを追加します。タグ付きインターフェイスは、複数の VLAN に 1 つのインターフェイスを割り当てる場合に使用します。「*VLAN（仮想ローカルエリアネットワーク）*」も参照してください。

チェーン

チェーンは一連のフィルタ基準で、IP アドレスへのアクセスを制限します。チェーン内での基準の順序によって、一般的な基準を先に適用し、詳細な基準をチェーンの最後に適用するなど、フィルタの適用方法を規定します。

チャンク

「*HTTP チャンク*」を参照してください。

データグループ

データグループは、AOL クライアントの IP アドレスのセットなど、関連する要素のグループです。データグループを **matchclass** コマンドまたは **contains** 演算子で指定すると、iRule 式内で複数の値を引数として並べる必要がなくなります。

デフォルト VLAN

BIG-IP では、各インターフェイスに 1 つずつ、2 つのデフォルト VLAN が設定されています。デフォルト VLAN には、**internal**（内部）と **external**（外部）があります。「*VLAN（Virtual Locatl Area Network: 仮想ローカルエリアネットワーク）*」も参照してください。

デフォルトプロファイル

デフォルトプロファイルは、BIG-IP がデフォルト設定値で提供するプロファイルです。デフォルトプロファイルをそのまま使用することも、変更することもできます。デフォルトプロファイルは、カスタムプロファイルを作成するときの親ファイルに指定することもできます。デフォルトプロファイルの作成や削除はできません。「*プロファイル*」、「*カスタムプロファイル*」も参照してください。

デフォルトルート

デフォルトルートは、ルーティングテーブルで指定されたルートに、宛先アドレスや転送されるパケットのネットワークに一致するものがない場合にシステムが使用するルートです。

デフォルトワイルドカードバーチャルサーバ

デフォルトワイルドカードバーチャルサーバは、IP アドレスとポート番号として、**0.0.0.0:0** または ***:*** または **"any":"any"** が設定されています。このバーチャルサーバは、設定で定義されたその他のバーチャルサーバに一致しないすべてのトラフィックを受信します。

透過ノード

透過ノードは、BIG-IP を含め、他のネットワークデバイスからはルータとして認識されます。

ドメイン名

ドメイン名は、1 つ以上の IP アドレスに関連付けられた一意の名前です。ドメイン名は、URL で使用して特定の Web ページを指定します。たとえば、**http://www.siterequest.com/index.html** という URL では、ドメイン名は **siterequest.com** です。

トラフィッククラス

トラフィッククラスを使えば、ユーザ定義の基準（送信元 IP アドレスや宛先 IP アドレスなど）に従い、トラフィックを分類することができます。トラフィッククラスによって、BIG-IP では分類に基づきトラフィックフローを調整できるようになっています。

トランク

トランクは、2つ以上のインターフェイスとケーブルを組み合わせて、1 つのリンクとして構成したものです。

内部 VLAN

内部 VLAN は、BIG-IP のデフォルト VLAN です。基本設定では、この VLAN の管理ポートは開いています。通常の設定では、これは内部サーバからの接続を処理するネットワークインターフェイスになります。

名前解決

名前解決は、ネームサーバがドメイン名要求を IP アドレスと一致させ、情報を解決を要求するクライアントに送信するプロセスです。

認証

認証は、ユーザが BIG-IP にログオンしようとするときに、ユーザの身元を検証するプロセスです。

認証 iRule

認証 iRule は、BIG-IP 付属またはユーザ作成の iRule で、BIG-IP で PAM 認証モジュールを実装するために必要です。「*iRule*」、「*PAM (Pluggable Authentication Module)*」も参照してください。

認証局（CA）

認証局は、外部の信頼された機関で、SSL ネットワークトラフィックの認証を取得するための資格情報として使用する署名済みデジタル証明書を、要求したコンピュータシステムに発行します。

認証プロファイル

認証プロファイルは設定ツールで、PAM 認証モジュールを実装するときに使用します。認証プロファイルで実装できる認証モジュールのタイプは、LDAP、RADIUS、TACACS+、SSL クライアント認証 LDAP、および OCSP です。「*PAM（Pluggable Authentication Module）*」も参照してください。

認証モジュール

認証モジュールは、ユーザが作成した PAM モジュールで、クライアントトラフィックの認証または承認を実行します。「*PAM（Pluggable Authentication Module）*」も参照してください。

ネットワークバーチャルサーバ

ネットワークバーチャルサーバは、IP アドレスのホスト部分にビットがない（IP アドレスのホスト部分が **0**）バーチャルサーバです。ネットワークバーチャルサーバには、2 種類あります。クライアントトラフィックを宛先 IP アドレスの範囲に基づいてダイレクトするものと、クライアントトラフィックを BIG-IP が認識しない特定の宛先 IP アドレスに基づいてダイレクトするものです。

ノード

ノードは、BIG-IP の論理オブジェクトで、ネットワークの物理リソースの IP アドレスを特定します。ノードは、プールメンバとモニタに直接関連付けられます。「*プールメンバ*」、「*モニタ*」も参照してください。

ノードアドレス

ノードアドレスは、1 つ以上のノードに関連付けられる IP アドレスです。この IP アドレスは、ネットワークサーバの実際の IP アドレスであっても、ネットワークサーバ上のエイリアス IP アドレスであっても構いません。「*プールメンバ*」、「*モニタ*」も参照してください。

ノードエイリアス

ノードエイリアスは、BIG-IP が複数のノードの状態を検証するのに使用するノードアドレスです。BIG-IP はノードエイリアスを使用してノードの状態をチェックする際に、ノードエイリアスに ping を実行します。BIG-IP は ping への応答を受信すると、ノードエイリアスに関連付けられたすべてのノードを **up** とマークします。BIG-IP は ping への応答を受信しない場合は、ノードエイリアスに関連付けられたすべてのノードを **down** とマークします。

ノードの状態

ノードの状態は、ノードが **up** で接続を受信できる状態か、または **down** で利用できない状態であるかを示します。BIG-IP はノードに対する ping の実行とヘルスチェック機能を使用して、ノードの状態を確認します。

ノードポート

ノードポートは、特定のノードがホストするポート番号またはサービス名です。

パーシステンス

「*接続パーシステンス*」または「*セッションパーシステンス*」を参照してください。

パーシステンスプロファイル

パーシステンスプロファイルは、特定のタイプのセッションパーシステンスを実装する設定ツールです。パーシステンスプロファイルタイプの例として、Cookie パーシステンスプロファイルがあります。

バースト

バーストとは、レートシェーピングの一面で、トラフィックフローのレートが定義されたベースレートを超過すると発生します。

パーティション

パーティションは、定義された BIG-IP システムオブジェクト のセットを含む、ユーザにより 作成される論理コンテナです。パーティションを使用して、BIG-IP へのユーザアクセスを制御します。「*ユーザロール*」も参照してください。

パイプライン

パイプラインは、HTTP/1.1 の機能で、以前の要求に対する応答がサーバから受信されていなくても、クライアントが要求を行えるようにします。

パケットレート

パケットレートは、サーバが処理する秒当たりのデータパケット数です。

パッシブモニタリング

パッシブモニタリングはクライアントリクエストの一環として行われます。この種のモニタリングは、一定時間内に試みられる接続やデータ要求の回数を指定して、プールメンバの健全性をチェックします。「*アクティブモニタリング*」と「*シンプルモニタリング*」もあわせて参照してください。

ハッシュパーシステンス

ハッシュパーシステンスによって、既存の iRule に基づいてパーシステンスハッシュを作成できます。

パフォーマンスモニタ

パフォーマンスモニタは、統計情報を収集してターゲットデバイスの状態を確認します。

標準 SNAT

標準 SNAT は、設定ユーティリティの SNAT 画面を使用して実装した SNAT です。「*SNAT（セキュアネットワークアドレス変換）*」および「*インテリジェント SNAT*」も参照してください。

プール

プールは、プールメンバの論理グループです。BIG-IP は、プール設定時に選択したロードバランシング方式とパーシステンス方式に従って、プールメンバに対する要求をロードバランスします。「*ノード*」、「*プールメンバ*」も参照してください。

プールメンバ

プールメンバは、ロードバランシングプールのメンバの 1 つです。プールメンバ名は、ノード IP アドレスとサービス番号を示します。「*ノード*」も参照してください。

フェイルオーバー

フェイルオーバーは、アクティブユニット上でソフトウェア障害またはハードウェア障害が検出された場合に、冗長システムのスタンバイユニットが処理を引き継ぐプロセスです。

フェイルオーバーケーブル

フェイルオーバーケーブルは、冗長システムの 2 つのユニットを直接接続します。

フェイルバック

フェイルバックは、アクティブユニットが、以前に失敗したが今は使用できるようになったユニットに処理を渡すプロセスです。

フォワーディングバーチャルサーバ

フォワーディングバーチャルサーバは、負荷を与えるプールメンバを持たないバーチャルサーバです。バーチャルサーバは、クライアントリクエストで指定されている宛先 IP アドレスにパケットを直接転送するだけです。「*バーチャルサーバ*」も参照してください。

フローティングセルフ IP アドレス

フローティングセルフ IP アドレスは、VLAN の追加セルフ IP アドレスで、BIG-IP 冗長システムの両方のユニットで共有されるアドレスです。

プロトコルプロファイル

プロトコルプロファイルは、FastL4、TCP、UDP、OneConnect、RTSP トラフィックの動作を制御するために作成するプロファイルです。

プロファイル

プロファイルは、ネットワークトラフィックの動作を定義する設定を含む、設定ツールです。BIG-IP には、FastL4、HTTP、TCP、FTP、SSL、RTSP トラフィックを管理し、パーシステンスおよびアプリケーション認証を実装するプロファイルがあります。

プロファイル設定

プロファイル設定は、プロファイル内の設定属性で、値が関連付けられています。プロファイル設定を行って、BIG-IP がトラフィックを管理する方法をカスタマイズできます。

プロファイルタイプ

プロファイルタイプは、特定の目的に使用できるプロファイルのカテゴリです。プロファイルタイプの 1 つに HTTP プロファイルがありますが、これはHTTPネットワークトラフィックの管理用に設定します。

ヘッダ

ヘッダとは、アプリケーションが伝送中のペイロードの前に付ける補足データのことです。ヘッダには、その後に続くペイロードをどう扱えばいいかの情報が含まれます。

ヘルスチェック

ヘルスチェックは BIG-IP の機能で、ノードが **up** か **down** かを判定します。ヘルスチェックは、ヘルスモニタで実装されます。「*ヘルスモニタ*、「*ECV*」、「*EAV*」、「*外部モニタ*」もあわせて参照してください。

ヘルスモニタ

ヘルスモニタは、ノードが **up** で指定されたサービスを実行しているか確認します。ノードがこのチェックに合格しないと、**down** と判定されます。チェックするサービスごとに、異なるモニタがあります。「*ヘルスチェック*」、「*EAV*」、「*ECV*」、「*外部モニタ*」もあわせて参照してください。

ポート

ポートは、ホストがサポートする特定のサービスに関連付けられた番号で表されます。ポート番号と対応するサービスの一覧については、「サービス」および「ポート」の項目を参照してください。

ポート固有のワイルドカードバーチャルサーバ

ポート固有のワイルドカードバーチャルサーバは、**0** 以外のポート番号を使用するワイルドカードバーチャルサーバです。「*ワイルドカードバーチャルサーバ*」も参照してください。

ポートミラーリング

ポートミラーリングは、ポートまたはポートの集合から、検出デバイスが接続された独立した1つのポートにトラフィックをコピーする機能です。

ホストバーチャルサーバ

ホストバーチャルサーバは、インターネット Web サイトや FTP サイトなど、特定のサイトを表すバーチャルサーバで、プールのメンバであるコンテンツサーバに向かうトラフィックのロードバランシングを実行します。

モニタ

BIG-IP はモニタを使用して、ノードが **up** か **down** かを判定します。モニタには複数の種類があり、さまざまな手法を使用してサーバまたはサービスの状態を判定します。モニタをノード、プール、および個々のプールメンバに関連付けることができます。「*ノード*」、「*プール*」、「*プールメンバ*」も参照してください。

モニタインスタンス

モニタインスタンスは、ヘルスモニタがプールメンバまたはノードに関連付けられるときに作成します。ヘルスチェックを実際に実行するのはモニタインスタンスで、モニタではありません。

モニタテンプレート

モニタテンプレートは、BIG-IP が使用する内部メカニズムで、事前に設定されたモニタがない場合にカスタムモニタのデフォルト値を提供します。

モニタの関連付け

モニタの関連付けは、ユーザが作成する、ヘルスモニタまたはパフォーマンスモニタと、プール、プールメンバ、ノードとの関連付けです。

ユーザ設定セット（UCS）

ユーザ設定セットは、BIG-IP のシステム設定データで作成するバックアップファイルです。UCS を作成すると、BIG-IP はファイル名に拡張子 **.ucs** を割り当てます。「*アーカイブ*」も参照してください。

ユーザロール

ユーザロールは、BIG-IP ユーザアカウントに割り当てる、アクセス権限のタイプとレベルです。ユーザロールを割り当てることで、BIG-IP のシステム管理者が BIG-IP の設定を表示して変更できる範囲を制御できます。

ユニバーサルパーシステンス

ユニバーサルパーシステンスによって、パケット内のあらゆる文字列を永続的に維持できます。また、プールメンバを直接選択して、パーシステンスにすることもできます。

予測方式

Predictive 方式は Observed 方式と同じ動的ロードバランシング方式であり、その時の接続数に基づきノードの評価が行われます。ただし、Predictive 方式では、BIG-IP は経時的なランキングの傾向を分析して、

ノードのパフォーマンスが現在向上しているか劣化しているかを判断します。より優れた（現在劣化しているのではなく向上している）パフォーマンスランキングを持つノードが、より高い比率の接続を受信します。

ラストホップ

ラスト ホップは、接続が BIG-IP に到達するまでの最後のホップです。BIG-IP で、パケットを送信元のデバイスに自動的に送り返すためのラスト ホップを決定することができます。また、ラスト ホッププールのメンバにすることで、ラスト ホップを手動で指定することもできます。

リモート管理 IP アドレス

リモート管理 IP アドレスは、BIG-IP で Telnet や SSH などのシェル接続を可能にする IP ドレスです。

リンクアグリゲーション

リンクアグリゲーションは、複数のリンクを結合し、1 つのリンクとして処理することで帯域幅を増加させるプロセスです。リンクアグリゲーションは、トランクを作成すると発生します。「*トランク*」、「*LACP（Link Aggregation Control Protocol）*」も参照してください。

ルータ

ルータは、レイヤ 3 ネットワーキングデバイスです。ネットワークで VLAN が定義されていない場合、ルータがブロードキャストドメインを定義します。

ループバックアダプタ

ループバックアダプタは、実際のネットワークカードとは関連付けられていないソフトウェアインターフェイスです。nPath ルーティング設定では、ループバックアダプタをサーバに設定する必要があります。

レイヤ 1 からレイヤ 7

レイヤ 1 からレイヤ 7 は、OSI（Open System Interconnection）モデルの 7 層を示します。つまり、レイヤ 2 はデータリンク層、レイヤ 3 は IP 層、レイヤ 4 はトランスポート層（TCP および UDP）を表します。レイヤ7はアプリケーション層で、HTTPおよびSSLなどのトラフィックを処理します。

レイヤ 2 転送テーブル

レイヤ 2 転送テーブルは、ネットワークデバイスの MAC アドレスを、そのデバイスがアクセスできる BIG-IP システムインターフェイスに関連付けます。BIG-IP では、各 VLAN には独自のレイヤ 2 転送テーブルがあります。

レートクラス

レートフィルタは、設定ユーティリティまたはコマンドラインユーティリティから作成できます。レートクラスをレートフィルタに割り当てると、レートクラスは、レートフィルタで許可されるトラフィックの量を判定します。「*レートシェーピング*」も参照してください。

レートシェーピング

レートシェーピングは、拡張 IP フィルタの一種です。レートシェーピングは同じ IP フィルタ方式を使用しますが、レートクラスを適用して、許可されたネットワークトラフィックの量を判定します。「*レートクラス*」も参照してください。

レスポンダオブジェクト

「*OCSP レスポンダオブジェクト*」を参照してください。

ローカルトラフィック管理

ローカルトラフィック管理は、イントラネットなどのローカルエリアネットワーク（LAN）から送受信されるネットワークトラフィックを管理するプロセスです。BIG-IP® LTM を使用して、ローカルトラフィックを管理できます。

ロードバランシングバーチャルサーバ

ロードバランシングバーチャルサーバは、クライアントトラフィックをロードバランシングプールにダイレクトするバーチャルサーバで、バーチャルサーバの基本タイプです。「*バーチャルサーバ*」も参照してください。

ロードバランシングプール

「*プール*」を参照してください。

ロードバランシング方式

接続をロードバランシングプールに分散させる方法を決定する特定の方式です。

ワイルドカードバーチャルサーバ

ワイルドカードバーチャルサーバは、IP アドレスとして、**0.0.0.0**、*****、または **"any"** を使用するバーチャルサーバです。ワイルドカードバーチャルサーバは、ローカルネットワークの外部の宛先への接続要求を受理します。ワイルドバーチャルサーバは、透過ノードモードの設定の場合にのみ使用できます。

索引

記号

/config/eav file A-45
/etc/bigip.comf ファイル 17-19

A

Accept-Encoding ヘッダ 6-11
Accounting Bug の設定 10-12
Active Directory サーバ、認証用 10-4
Adaptive 圧縮
  6400、6800、8400 での設定 6-25
  概要 6-23
admin アカウント 10-5
[AIA] フィールド 10-30
Apache バリアント 7-9
Application Security Manager 8-13
ARP 要求 2-20
ASN.1 DER 形式 C-3
[AuthorityInfoAccess] フィールド 10-30

B

[Base Rate] 設定、設定 16-5
BEA WebLogic アプリケーション 1-18
BIGipCookie Cookie 名 7-9
BIGipServer<pool_name> Cookie 名 7-8, 7-9
BIG-IP オブジェクトの状態
  表示 1-16
bigpipe -? コマンド 17-5, 17-11
Bind DN の設定 10-5
Bind Password の設定 10-5
Bind Time Limit の設定 10-5
[Burst Size] 設定、設定 16-6

C

[Ceiling Rate] 設定、設定 16-5
Certificate Revocation List Distribution Point
  CRLDP モジュールを参照
Check SSL Peer の設定 10-6
Client ID の設定 10-12
clientside iRule コンテキスト 17-10
CMP 2-8
Configuration の設定 10-7, 10-13, 10-42
Confirm Bind Password の設定 10-5
Confirm Secret の設定 10-11, 10-39
contains 演算子 17-16
Content-Encoding ヘッダ 6-11
Content-Type 応答 6-16
Cookie
  HTTP Cookie パーシステンスも参照
Cookie Hash モード 7-10
Cookie、HTTP Cookie Rewrite 7-9
Cookie 値 7-10
Cookie テンプレート 7-10
Cookie の暗号化と復号化 6-9
Cookie パーシステンス
  HTTP Cookie パーシステンスも参照
  定義 7-7
Cookie プロファイルの設定 7-7
Cookie 名 7-8
CPU 使用率
  圧縮 6-21
CPU メトリック、収集 A-53
CRL
  LDAP サーバ 10-38
  OCSP も参照
  クライアント側 SSL 9-29
  欠点 10-29
  作成 C-3
  生成
  説明 9-5
  タイムアウト 10-40
  バイパス 10-30
  表示 C-4
CRLDP 認証モジュール C-3
CRLDP モジュール、定義 10-38
CRL 配布点 10-38
CRL ファイル 9-29

D

Debug Logging の設定 10-6, 10-12
deflate 圧縮方式 6-17
destaddr プロファイル設定 7-11, 7-12
[Direction] 設定、設定 16-8
dirname ベースのアドレス C-1
DNS 参照 A-23
Dynamic Ratio モード
  RealSystem Server の設定 B-1
  WMI の設定 B-3
  説明 4-12

F

Fast HTTP プロファイル設定 8-7
Fast L4 プロファイル設定 8-2
Fastest モード、説明 4-12
Filter の設定 10-5
FQDN、リダイレクション 6-5
FTP プロファイルの設定 6-38

G

genconf および genkey ユーティリティ 9-4
Group DN の設定 10-6
Group Member Attribute の設定 10-6
gzip 圧縮、メモリレベル 6-19
gzip 圧縮方式 6-17

H

Hosts の設定 10-5
Host の設定、RADIUS オブジェクト 10-10
HTTP Class プロファイルの設定 8-10
HTTP Cookie Passive モード 7-9
HTTP Cookie Insert 方式 7-8
HTTP Cookie Rewrite 方式 7-9
HTTP Cookie パーシステンス 7-7, 7-8
HTTP Location ヘッダ 6-8
HTTP Redirect Rewrite 設定 6-5
HTTP/1.0 圧縮 6-20
httpd.conf ファイル、Cookie 7-9
HTTP 圧縮設定 6-13, 6-32
HTTP 応答、圧縮 6-12

HTTP オブジェクト
　格納 6-28
HTTP 書き換え、例 6-8
HTTP コンテンツ
　検査 6-5
HTTP 承認の失敗 6-4
HTTP データ圧縮
　説明 6-11
　有効化 6-15
HTTP トラフィック管理 6-1
HTTP パイプライニング
　パイプライニングを参照
HTTP プロファイル
　説明 6-1
　デフォルトおよびプロファイル 5-6
HTTP プロファイル設定、設定 6-2
HTTP プロファイルの共有 1-20
HTTP ヘッダ
　検査 6-5
　挿入 6-5
HTTP ヘッダコンテンツ、消去 6-6
HTTP メソッド、キャッシュ 6-29
HTTP モニタ A-64
HTTP リクエスト、リダイレクト 17-5
HTTP リダイレクション
　書き換え 6-8
　プールの選択 17-5
　例 17-5

I

ID、信頼 9-6
IDEA 暗号スイート 9-17
Idle Timeout の設定
　LDAP プロファイル 10-7
　RADIUS プロファイル 10-13, 10-42
Ignore Unknown User の設定 10-6
ignore オプション 9-26
IIS アプリケーション 1-18
IMAP モニタ A-24
IP アドレス
　Cookie 7-9
　NAT に指定 13-6
　SNAT プールへのマッピング 14-6
　一致 2-5, 2-6
　共有 2-2
　クライアント 7-19
　パーシステンス 7-20
　バーチャルサーバ 2-5
　変換 2-6
　リダイレクション 6-5
IP アドレスデータグループ 17-17
IP アドレスの宛先 2-6
iRule
　HTTP ヘッダの挿入 9-6
　SNAT 14-6
　SSL トラフィック管理 9-1
　暗号 9-5
　作成 17-7
　定義 17-1
　認証 10-2
　ネットワークマップ 1-17
　バーチャルサーバ 17-7
　バーチャルサーバとの関連付け 17-8
　割り当て 2-23, 17-11
iRule イベント宣言 17-2
iRule イベントタイプ 17-10
iRule 演算子 17-3
iRule コマンドの種類 17-3
iRule ステートメントコマンド 17-12
iRule の設定
　LDAP プロファイル 10-7
　RADIUS プロファイル 10-13, 10-42
iRule の前提条件 17-9
iRule の例
　HTTP リダイレクション iRule 17-5
iRule 評価、制御 17-9
iRule 要素 17-2
[Issuer] フィールド 10-30

K

Keep-Alive サポート、追加 1-4

L

L2 フォワーディングバーチャルサーバ、定義 2-2
L3 Address モード 4-13
LDAP 10-4
LDAP CRL 分散ポイント C-2
LDAP Version の設定 10-5
LDAP 構成オブジェクト、作成 10-5
LDAP サーバ 10-38
　トラフィック認証 10-21
　トラフィック認証と承認 10-16, 10-21, 10-44
　認証、承認 10-4
LDAP 承認の基準 10-23
LDAP 設定済みモニタ A-22
LDAP データベース、検索 10-22
ldap デフォルトプロファイル、変更 10-7
LDAP 認証、概念 10-22
LDAP 認証パラメータ、一覧 10-24, 10-34
LDAP プロファイル
　作成 10-7
　バーチャルサーバ 10-9
LDAP ベースディレクトリ名 10-39
LDAP モジュール、定義 10-1
LDAP モニタ A-22
Least Connections モード、説明 4-12
Least Sessions モード 4-13
Lightweight Directory Access Protocol モジュール
　LDAP モジュールを参照、定義
Location ヘッダ 6-8
Login Attribute の設定 10-5

M

Manual Resume 属性 3-9, 12-12
matchclass コマンド 17-16
MD5 ハッシュ 9-20
Microsoft Exchange Outlook Web Access アプリケーション 1-18
Microsoft IIS アプリケーション 1-18
Microsoft SharePoint アプリケーション

SharePoint アプリケーション 1-18
min active members 値 4-14
ModSSL 方式のエミュレーション、有効化および無効化 9-20
MSRDP パーシステンス
旧プラットフォーム 7-15
有効化 7-14, 7-15
MSRDP プラットフォームの要件 7-14
MSRDP プロファイルの設定 7-15

N

Nagle's algorithm 8-17, 8-19, 8-20
NAT
オープンポート 13-2
設定 13-6
定義 13-1
例 13-3, 13-4
NAT アドレス
共有 13-3
NAT 接続
追跡 13-3, 14-1
NAT 設定 13-5, 13-6
nntp モニタ A-26
NT Lan Manager
NTLM プロファイルを参照
NTLM も参照
定義 11-4
NTLM プロファイル 2-16
目的 11-4

O

Observed モード、説明 4-13
OCSP 構成オブジェクト、作成 10-32, 10-34
OCSP 前提条件 10-31
OCSP プロトコル
説明 10-29
定義 10-30
OCSP プロファイル
作成 10-35
バーチャルサーバ 10-37
OCSP モジュール 10-2
OCSP レスポンダ
CA 10-31
選択 10-30
OCSP レスポンダ定義、選択 10-30
OCSP レスポンダオブジェクト
作成 10-32
定義 10-23
OneConnect
受信 SNAT 14-2
OneConnect、有効化 6-7
OneConnect プロファイル設定 11-2
Online Certificate Status Protocol モジュール
OCSP モジュールを参照
OpenSSL 0.9.8.x C-1
OpenSSL Web サイト 9-17
openssl コマンド 9-29
openssl ユーティリティ C-1, C-2, C-3, C-4
Oracle Application Server アプリケーション 1-18
Outlook Web Access アプリケーション 1-18

P

Packet Velocity ASIC 8-2
PAM、定義 10-1
PAM 認証モジュール、リスト 1-5
PAM モジュール 10-1
Parent Profile の設定 10-7
[Parent Class] 設定、設定 16-8
Passive モード 7-9
PEM 形式 9-10, 9-15
PEM 形式への変換 C-4
PFIFO、定義 16-9
PKCS12 ファイルの作成 C-1
Platform Guide 1-22
Pluggable Authentication Module
PAM モジュールを参照
PAM を参照
[Pools] 画面、表示 1-12
pop3 モニタ A-33
Predictive モード、説明 4-13
Privacy Enhanced Mail 形式
PEM 形式を参照
Protocol Security Manager 2-16
PSM
SMTP プロファイル 2-16
PVA ハードウェアの高速化 8-2, 8-5

Q

QoS プールの属性 4-10
QoS レベル
設定 4-10
[QueueDiscipline] 設定、設定 16-9

R

RADIUS アカウンティング A-67
RADIUS 構成オブジェクト、作成 10-11, 10-40
RADIUS サーバ
トラフィック認証 10-10
認証 10-10
RADIUS サーバオブジェクト、作成 10-10, 10-39
RADIUS プロファイル
作成 10-13, 10-41
バーチャルサーバ 10-43, 10-15
変更 10-42
RADIUS モジュール、定義 10-1
RADIUS モニタ A-35
RAM キャッシュ機能 6-28
RAM キャッシュ仕様 6-28
Ratio 方式、説明 4-11, 4-12
Real Time Streaming Protocol（RTSP）
RTSP プロトコルを参照
RealSystem Server、ロードバランシングの設定 B-1
Remote LDAP Tree の設定 10-5
request オプション 9-26
require オプション 9-26
Retries の設定 10-12
Reverse 設定 12-12
Rewrite モード 7-9
Round Robin モード、説明 4-11
RPC モニタ A-37, A-42
RSA キー C-4

RTCP Port 6-45
RTP Port 6-45
RTSP トンネリング 6-44
RTSP プロトコル
　定義 6-44, 6-47
RTSP プロファイル 2-16
　TCP 6-44
　バーチャルサーバ 6-44, 6-45
RTSP プロファイル設定 6-45
RTSP プロファイルの設定 6-44, 6-47

S

SAP アプリケーション 1-18
SASP モニタ A-42
Scripted モニタ A-44
SCTP ストリーム 8-22
SCTP プロトコル
　定義 8-22
SCTP プロファイルの設定 8-22
Search Time Limit の設定 10-5
[Search] ボックス
　ネットワークマップ 1-13
　プロファイルリストのフィルタリング 5-11
Secret の設定 10-11
serverside iRule コンテキスト 17-10
Service Port の設定
　LDAP モジュール 10-5
　RADIUS オブジェクト 10-10, 10-39
Session Directory、MSRDP パーシステンス 7-15
Session Directory サービス 7-15
Session Initiation Protocol 6-40, 7-17
SFQ キュー方式
　定義 16-9
Simple Mail Transport Protocol 2-16
SIP パーシステンス 6-43, 7-17
SIP パーシステンス、定義 7-17
SIP プロファイル設定 6-40
SIP プロファイルのセッション終了基準 6-42
SIP プロファイルの設定 6-40, 7-17
SIP ヘッダサイズの指定 6-42
SIP ヘッダの取り扱い 6-42
SIP メッセージへの Record-Route ヘッダの挿入 6-43
SIP メッセージへの Secure Via ヘッダの挿入 6-43
SIP メッセージへの Via ヘッダの挿入 6-43
SIP モニタ A-46
SMB モニタ A-49
SMTP プロファイル 2-16
SMTP モニタ A-51
SNAT
　受信 SNAT もあわせて参照
　OneConnect 14-2
　仮想バウンスバック 14-2
　定義 14-1
　例 14-4, 14-18
SNAT automap
　定義 14-5
SNAT、有効化および無効化 4-8
snatpool コマンド 14-9, 17-6
SNAT アドレス
　デフォルトゲートウェイ 14-2
SNAT 接続
　追跡 14-1
SNAT と NAT の比較 14-1
SNAT の動作 14-4
SNAT プール
　作成 14-8
　定義 14-6
　バーチャルサーバ 2-18
　バーチャルサーバへの割り当て 14-13
　パーティション 14-7
SNAT プールの設定 2-18, 14-7
snmp_dca_base template A-55
SNMP DCA Base モニタ B-10
SNMP DCA モニタ B-10
SOAP モニタ A-56
SQL Enterprise Manager A-28
SQL ベースのサービス、サービスのチェック A-26
SQL ベースのサービスのチェック、トラブルシューティング A-28
SSL CA Certificate の設定 10-6
[SSL Certificates] 画面 9-7
SSL Ciphers の設定 10-6
SSL Client Certificate の設定 10-6
SSL Client Key の設定 10-6
SSL OCSP 構成オブジェクト、作成 10-34
SSL、クライアントへの認証 9-15
SSLv2 プロトコル 9-19
SSLv3 プロトコル 9-19
SSL 暗号化および復号化 9-5
SSL キー、管理 9-7
SSL キー暗号化 9-15
SSL 機能の要約 9-2
SSL クライアント証明書 LDAP 構成オブジェクト 10-24
SSL クライアント証明書 LDAP プロファイル 10-26, 10-28
SSL クライアント証明書 LDAP モジュール、定義 10-2
SSL 検証、チェーンファイル 9-4
SSL シャットダウン 9-23
SSL 承認
　説明 9-3
SSL 証明書、管理 9-7
SSL セッション、ネゴシエーション 9-28
SSL セッションキャッシュのサイズ 9-21
SSL セッションキャッシュのタイムアウト 9-20, 9-21
SSL セッションの再ネゴシエーション、強制 9-22
SSL 接続、シャットダウンアラート 9-23
SSL 設定作業 9-1
SSL タイムアウト期間 9-21
SSL 認証
　証明書の失効 10-29, 10-38
SSL 認証機能 9-2, 9-25
SSL の設定 10-6
SSL パーシステンス、定義 7-20
SSL パーシステンスタイプ 9-6
SSL パーシステンスプロファイルの設定 7-21
SSL パスフレーズ
　暗号化 9-15
　キー暗号化 9-15
SSL プロトコルバージョン、設定 9-19

SSL プロファイル
一覧 5-2
SSL プロファイル、定義とリスト 9-1
SSL プロファイルタイプ 9-1, 9-11
SSL プロファイル名、指定 9-14
SSL 問題回避策、設定 9-17
Statistics プロファイル 11-6
Stochastic Fair Queue
SFQ キュー方式を参照
Stream Control Transmission Protocol (SCTP)
SCTP プロトコルを参照
Stream プロファイル 11-7
SYN フラッド攻撃、防止 1-5
SYSLOG デバッグ 10-6

T

TACACS+ 構成オブジェクト、作成 10-16, 10-45
TACACS+ サーバ、トラフィック認証 10-16
TACACS+ プロファイル
作成 10-18, 10-46
バーチャルサーバ 10-20, 10-48
TACACS+ モジュール、定義 10-1
Tcl 構文 17-2
TCP ストリーム 8-22
TCP 接続、シャットダウンアラート 9-23
TCP モニタ A-64
Timeout の設定、RADIUS オブジェクト 10-11
TLSv1 プロトコル 9-19
TMM サービス
リモート認証 10-1
TMOS Management Guide for BIG-IP Systems 1-22
Tools Command Language の構文 17-2
ToS フィールド
キューイング 16-9
ToS プールの属性 4-9
ToS レベル、設定 4-9
Transparent 設定 12-11
transparent モード 4-8, 4-18
Type of Service フィールド
ToS フィールドを参照

U

UC Davis エージェント A-53, B-10
UDP プロファイル設定 8-21
UDP モニタ A-64
UIE、定義 17-1
UIE 関数コマンド、一覧 17-13
UIE コマンド、定義 17-3
Uniform Resource Identifier（URI） 10-38
Universal Inspection Engine、定義 17-1
URI
書き換え 6-8
包含と除外 6-12
リダイレクション 6-8
URI 指定応答、管理 6-15
URI パス、リダイレクト 6-4
URL チェック 10-30

V

Vary ヘッダ
挿入 6-19
有効化と無効化 6-19
VLAN グループ、作成 2-12
VMware VDI アプリケーション 1-18

W

WAP モニタ A-66
Warning Logging の設定 10-6
WebAccelerator モジュール 8-13
Web サーバ、Cookie 生成 7-10
[Welcome] 画面 1-25
when キーワード 17-11
Windows 2000 Server エージェント A-53, B-10
Wireless Application Protocol モニタ
WAP モニタを参照
WMI、Dynamic Ratio ロードバランシングの設定 B-3
WMI モニタ A-66

X

X-Forwarded-For ヘッダ 6-10

あ

アカウンティング、RADIUS A-67
圧縮
CPU 使用率 6-21
RAM キャッシュ機能 6-28
圧縮、有効化 6-15
圧縮設定 6-13, 6-32
圧縮戦略
概要 6-22
説明 6-23
圧縮統計情報の説明 6-26
圧縮バッファサイズ 6-17
圧縮プロバイダの概要 6-22
圧縮プロファイル 6-34
宛先 IP アドレス
パーシステンス 7-10
変換 2-3, 13-1
宛先 IP 変換 2-3, 14-3
宛先アドレスのアフィニティパーシステンス 7-10
宛先アドレスの範囲、ダイレクト先 2-5
宛先アドレス範囲 2-5
アドレスデータグループ 17-17
アドレス変換 1-7, 13-1
アドレス変換設定 2-17
アプリケーションセキュリティクラス 8-13
アプリケーションテンプレート
使用 1-19
説明 1-18
表示と削除 1-20
アプリケーション導入ガイド 1-19
アプリケーショントラフィック
管理 5-1
設定 1-18
認証 10-1
アラートタイムアウト、指定 9-21

暗号、ヘッダでの指定 9-16, 9-19
暗号化
Cookie 6-9
説明 9-5
要約 9-3

い

イベント実行、終了 17-11
イベント宣言 17-9
イベントベースのトラフィック管理 17-9
インテリジェント SNAT 14-9
インラインデータグループストレージ 17-19

え

エージェントタイプ B-10
エラーコード 6-5
エンコード、チャンクおよびチャンク解除 6-6
演算子 17-3

お

応答
圧縮 6-12
チャンクおよびチャンク解除 6-6
応答の受け入れ 14-4
応答の受け取り拒否 2-3, 14-2
図解 14-3
オブジェクトの状態 1-15
重み付け、指定 4-17
親 HTTP プロファイル、指定 6-4
親 SIP プロファイル、指定 6-42
親プロファイル
定義 5-5
親レートクラス、帯域幅の借用 16-8
オンラインヘルプ 1-25

か

カーネルルーティングテーブル 2-20
外部データグループ 17-20
外部データグループストレージ 17-19
カスタム HTTP プロファイル 5-6
カスタムプロファイル 5-4, 5-5, 10-8
カスタムモニタ 12-4
仮想バウンスバック 14-2
関係演算子、リスト 17-3
管理パーティション
アプリケーションテンプレート 1-18

き

キー
管理 9-7
削除 9-10
キー暗号化 9-15
キータイプ 9-7
キーと証明書のアーカイブ、インポート 9-10
キーと証明書のペア、インポート 9-4
キーのアーカイブ、作成 9-11
キーの作成 C-1
キーのペア、インポート 9-10
キーワード、予約済み 12-9, A-1
期限切れのセッション ID 9-21
基準の指定 8-10
規則 1-23
逆変換 2-3
キャッシュ項目
一覧 6-28
クリア 6-31
キャッシュプロファイル 6-34

く

空白の Cookie
概要 7-9
挿入と検索 7-9
クエリコマンド、定義 17-3
組み込み配布点 C-1
クライアント、セキュアな接続 6-8
クライアント CRL パス、指定 9-29
クライアント CRL ファイル、指定 9-29
クライアント IP アドレス
接続の追跡 7-19
ヘッダへの挿入 6-5
保持 6-5
クライアント側 SSL プロファイル 9-1
クライアント側のセッションキャッシュのサイズ 9-21
クライアント側の接続
証明書の検証 9-15
信頼できる CA 9-4
クライアント側のプロファイル、暗号化と復号化 9-5
クライアント検証プロセス 9-4
クライアント証明書、要求 9-26
クライアント証明書の作成 C-1
クライアントトラフィック
サブネットのダイレクト 2-5
リダイレクト 4-1
レートクラス 16-8
クライアント認証方式 9-26
クライアントの信頼できる CA パス、説明 9-15
クライアントの信頼できる CA ファイル、説明 9-15
クライアントの接続、認証 10-4, 10-10
クライアントヘッダ、承認 9-6
クライアントリクエスト、承認 10-21
クラスタリングマルチプロセッシング
CMP を参照
グループベースの LDAP 承認 10-23
グループベースの承認 10-23
クローンプール 2-18
クローンプール、バーチャルサーバ 2-18

け

形式
PEM 形式を参照
検索アカウント 10-5
検索方法、LDAP データベース 10-22
検証
証明書の検証、クライアント側参照
証明書の検証、サーバ側も参照

検証の失敗 10-31
検証プロセス
　クライアント検証プロセスを参照
　検証も参照

こ

高需要オブジェクト 6-28
更新インターバル 10-40
高速化
　CMP 2-8
　PVA 8-5
　ハードウェアの高速化を参照
コネクションプーリング
　NTLM
　NTLM 接続 11-4
　X-Forwarded-For ヘッダも参照
　定義 1-4, 11-2
コネクションリミット
　CMP が有効になったシステム 2-9
　許可 2-20
　ノード 3-7, 4-17
子レートクラス 16-8
コンテキスト、iRule 17-10
コンテンツ
　検査 8-10
コンテンツ圧縮 6-28
コンテンツの切り替え
　カスタマイズ 17-1
　定義 1-4
コンテンツの検索 17-1

さ

サーバ、iRule による選択 17-4
サーバオブジェクト、定義 10-2
サーバ側 SSL プロファイル
　暗号化 9-5
　管理 9-2
　定義 9-1
サーバ側の SSL 接続
　証明書の検証 9-15
　信頼できる CA 9-4
サーバ側の検証、説明 9-4
サーバ側のセッションキャッシュのサイズ 9-21
サーバ側の接続 1-4
サーバ証明書の作成 C-2
サーバチェーンファイル、説明 9-4
サーバトラフィック、レートクラス 16-8
サーバ認証 C-2
サーバのアベイラビリティ 3-9, 12-12
サーバのアベイラビリティ、向上 2-1
サーバの過負荷 4-1
サーバ負荷 6-28
サービスのチェック、トラブルシューティング A-28
サービスプロファイル
　一覧 5-1
サービスプロファイル、リスト 5-1
最小ヘルスモニタ 3-7
サイズ、SSL セッションキャッシュ 9-21
サブネット
　サーバノード 14-2

し

識別名 10-5
識別名、指定 10-5
自己署名証明書、生成および要求 9-7
システムパフォーマンス、監視 1-7
失効
　CRL を参照
失効、証明書 9-5
失効状態
　チェック 10-30
　評価 10-29
送信元アドレスのアフィニティパーシステンス 7-19
自動エンコード 7-10
シャットダウンアラート 9-23
受信 SNAT
　OneConnect 14-2
　同じネットワーク 14-2
　図解 14-4
　定義 14-3
受信接続
　着信接続を参照
上限レート 16-6, 16-9
承認
　SSL 証明書の使用 9-3
　グループとロール 10-23
承認オプション、リスト 9-6
承認の失敗 6-4
承認パラメータ、一覧 10-24, 10-34
証明書
　CA からの要求 9-8
　インポート 9-10
　管理 9-7
　更新 9-9
　削除 9-10
　承認 10-21
　信頼 9-15
　生成およびインストール 9-4
　必要 9-26
　ヘッダとしての挿入 9-6
　目的 9-3
証明書検証の失敗 9-29, 10-31
証明書失効状態
　チェック 10-30
　評価 10-29, 10-30
証明書失効リスト
　CRL を参照
証明書情報の確認 C-4
証明書チェーンファイル 9-29
証明書とキーのペア、要求 9-7
証明書のアーカイブ、作成 9-11
証明書の検証 C-4
証明書の検証、クライアント側 9-4, 9-15
証明書の検証、サーバ側 9-4, 9-15
証明書の失効 C-3
　CRL を参照
証明書のステータス、表示 9-7
証明書の認証機能 9-2
証明書の発行者 9-7

証明書の比較、検索方法 10-22
証明書のマッピング、検索方法 10-22
証明書要求、送信 9-8
証明書要求方式 9-8
署名入りの証明書
　承認 10-21
　証明書も参照
侵入検知 2-18
シンプルパーシステンス 7-19
信頼できる CA、指定 9-4
信頼できる CA パスの名前、指定 9-15
信頼できる CA ファイルの名前、指定 9-15
信頼できる CA リスト、送信 9-28
信頼できる認証局のファイル 10-30

す

数式とエンコード 7-10
数値クラス 17-18
スタティックコンテンツ 6-28
スティッキーパーシステンス
　宛先アドレスのアフィニティパーシステンスも参照
スティッキーパーシステンスタイプ 7-10
ステータスアイコン
　ノード 3-9
　バーチャルサーバおよびアドレス 2-24
　プール 4-22
　プールメンバ 4-24
ステートメントコマンド
　指定 17-12
　定義 17-3
ストリーミングメディアプロトコル
スループット
　カスタマイズ 1-5
　実施 16-1
スループット制限 16-2
スループット制約、適用 16-8
スループットポリシー 16-1
スループットポリシー、実施 16-1
スループットレート 16-5

せ

正規表現
　データ圧縮 6-12
　トラフィック分類 6-11, 8-10
整数データグループ 17-18
セキュアネットワークアドレス変換
　SNAT を参照
セキュリティ侵犯、防止 1-5
セッション ID、挿入 9-6
セッションキャッシュのサイズ 9-21
セッションキャッシュのタイムアウト 9-21
セッションの共有 7-15
セッションの再ネゴシエーション、強制 9-22
セッションの認証 9-27
セッションパーシステンス 2-1
　iRule 17-15
　SSL 接続 9-6
セッションロス
　監視と検出 8-22
接続
　シャットダウンアラート 9-23
　証明書の検証 9-15
　信頼できる CA 9-4
　認証 10-4, 10-10
　プライオリティによる分散 4-14
接続キューイング 16-9
接続の終端 9-26
接続パーシステンス 6-7
接続分配 4-13
接続要求
　受信 2-5
設定
　プロトコルプロファイル 8-1
設定ユーティリティ
　Web ベースの使用 1-7
　Welcome 画面 1-25
　オンラインヘルプの概要 1-25
　設定ユーティリティ 1-7
　目的 1-7
設定用ワークシート 1-22
セルフ IP アドレス
　SNAT 14-5
セルフ IP アドレス、割り当て 2-12
線形空白、管理 6-10

そ

送信元 IP アドレス
　変換 13-1
送信元 IP 変換 2-3, 14-3
送信元アドレスのアフィニティパーシステンス 7-19
Insert モード、HTTP Cookie パーシステンス 7-8

た

ターミナルサーバの設定 7-15
帯域幅
　借用 16-8, 16-9
　補給 16-6
　保存 16-6
ダイナミック IP アドレス、パーシステンス 7-20
タイムアウト値 7-19
短命ポート 14-3

ち

チェーンファイル 9-4, 9-15
着信接続
　NAT 13-2
チャンク 6-6
中間 CA、信頼 9-15
中間者攻撃、防御 9-22

て

ディスクメトリック、収集 A-53
データ圧縮
　説明 6-11
　有効化 6-15
データグループ
　設定 17-16

保存 17-20
データグループサイズ 17-17
データグループストレージ 17-19
データグループタイプ 17-17
データグループメンバ 17-21
データ収集エージェント B-10
テストアカウント、作成 A-28
デフォルト HTTP プロファイル 5-6, 6-2
デフォルト LDAP プロファイル、変更 10-7
デフォルト RADIUS プロファイル 10-42
デフォルトゲートウェイ 14-2, 14-3
アドレス変換 2-3
デフォルトの圧縮値 6-13, 6-32
デフォルトのパーシステンスプロファイル 2-19
デフォルトのワイルドカードバーチャルサーバ、作成 2-12
デフォルトプロファイル 5-3
使用 5-3
要約 5-5
デフォルトルート 14-2
テンプレート
アプリケーションテンプレートも参照
アプリケーションの設定 1-18

と

透過デバイス、接続の受信 2-7
透過デバイスプール、作成 2-12
透過ノード 2-6, 2-7
統計情報
表示 2-8
統計情報、バーチャルサーバ 2-23
同時コネクションの制限 4-17
同時接続 2-20
同時接続の制限 3-7
匿名検索、許可 10-5
トラフィック
QoS レベル 4-10
管理 5-1
キューイング 16-9
認証 10-1
プライオリティによる分散 4-14
リダイレクト 6-4
トラフィッククラス
管理 15-3
作成 15-1
定義 15-1
トラフィッククラスの設定 15-2
トラフィッククラスは 15-1
トラフィッククラスリスト 2-17
トラフィックタイプ、管理 2-1
トラフィックのキューイング 16-9
トラフィックの高速化
CMP 2-8
Fast L4 プロファイルの使用 8-2
PVA 8-5
トラフィックの方向とレートクラス 16-1
トラフィックの複製 2-18
トラフィック複製 2-18
トラフィックフロー制限 16-5
トラフィックフロータギング 2-17
トラフィックフローレート 16-6, 16-8
トラフィック分類 8-10
トラフィックレート、バースト 16-6

に

認識されない宛先アドレス 2-5
認証、セッションごと 9-27
認証 iRule
定義 10-2
割り当て 10-9, 10-43
認証 iRules
割り当て 10-15
認証局 10-30
CA を参照
認証プロファイル
一覧 5-2
定義 10-2
割り当て 10-9, 10-15, 10-43
認証方式 9-26
認証モジュール
一覧 1-5
実装 10-2
定義とリスト 10-1
認証モジュールの種別 10-1
認証レベル 9-29

ね

ネットマスク、指定 2-12
ネットワーク IP アドレスデータグループ 17-17
ネットワークアドレス変換（NAT）
NAT を参照
ネットワークトラフィック
管理 5-1
認証 10-1
ネットワークバーチャルサーバ
定義 2-5
ネットワークバーチャルサーバのタイプ 2-5
ネットワークパフォーマンス、最適化 1-4, 1-5
ネットワークマップ
アプリケーションテンプレート 1-21
定義 1-16
表示 1-16, 1-17
ネットワークマップ機能
定義 1-13

の

ノード
コネクションリミット 3-7, 4-17
接続の受信 4-10
定義 3-1
トラフィックのダイレクト 7-20
プールメンバとして 4-7
ノード情報、Cookie 7-9
ノードのアベイラビリティ 3-9, 12-12

は

パーシステンス
iRule 17-1, 17-15
MSRDP プラットフォームの要件 7-14
SSL 接続 9-6

条件 7-5
必要性 7-1
平文のトラフィック 7-20
パーシステンスのタイマー 7-19
パーシステンスプロファイル
iRule 7-2
一覧 5-2
パーシステンスプロファイルの種類 7-4
バーストの制約事項 16-7
バースト保管域 16-6
バーチャルサーバ
Fast HTTP プロファイル 2-2
Fast L4 プロファイル 2-2
iRule 2-1, 17-11
RTSP プロトコル 6-44
SNAT 14-6, 14-9
宛先 8-10
削除 2-26
定義 2-1
ネットワークマップ 1-14
パーシステンス 7-5
表示 2-23
プロファイル 2-15, 5-12
無効化 2-12
バーチャルサーバアドレス、VLAN 2-12
バーチャルサーバ機能 2-1
バーチャルサーバの画面、表示 1-11
バーチャルサーバの設定
設定 2-14
バーチャルサーバのタイプ 2-5
バーチャルサーバの統計情報、表示 2-23
バーチャルサーバの動作 2-3
バーチャルサーバのプロパティ、設定 2-14
バーチャルサーバのマッピング、ワイルドカードの定義 2-13
バーチャルサーバのリソース
変更 2-22
割り当て 2-14
パーティション
iRule 17-7
SNAT 14-10
SNAT プール 14-7, 14-10
データグループ 17-16
ノード 3-2
バーチャルサーバ 2-11
プール 4-2
プロファイル 5-8
モニタ 12-8, 12-17
レートクラス 16-2
ハードウェアの高速化 8-2, 8-5
パイプライニング
定義
HTTP パイプライニング
íËã` 6-9
パイプライン
定義 1-4
パケット、キューイング、デキューイング 16-1
パケットスケジューリング手法 16-2
パケットスループット、実施 16-1
パケットの順序 16-9
パケットフィルタ 16-11
パケットレート、超過 16-5
パケットレート制限、指定 16-5
パケット、キューイング、デキューイング 16-9
パスフレーズ
SSL パスフレーズ参照
パスフレーズ暗号化 9-15
パスフレーズの長さ 9-15
パスワードの追加と削除 C-4
パターン文字列
トラフィック分類 6-11, 8-10
ハッシュパーシステンス、定義 7-12
発信接続
NAT 13-4
発信接続を参照
発信トラフィック、ToS レベル 4-10
バッファサイズ、圧縮 6-17
パフォーマンス
ロードバランシング 4-13
パフォーマンス（HTTP）バーチャルサーバ 2-2
パフォーマンス（レイヤ 4）バーチャルサーバ 2-2
パブリック IP アドレス
プライベートへのマッピング 13-1
パラメータ、LDAP 認証 10-21

ひ

ピア認証 9-15
表記規則 1-23
標準 SNAT 14-9
標準アドレス変換 14-2, 14-3
平文のトラフィック、ロードバランシング 7-20

ふ

ファイアウォール 2-6
ファイル、包含と除外 6-12
プール
iRule による選択 17-1, 17-4
SNAT または NAT 接続 4-8
削除 4-22, 4-26
定義 4-1
プール監視 4-7
プールステータス 4-22
プール設定、デフォルト値 4-16
プールとモニタの関連付け、管理 4-27
プール名 4-7, 14-11
プールメンバ
iRule による選択 17-4
サーバとして 4-7
追加 4-15
定義 1-12, 4-1
プールメンバステータス 4-24
プールメンバのアベイラビリティ 12-12
フォールバックパーシステンスプロファイル 2-19
フォールバックホスト 6-4, 6-5
フォワーディングバーチャルサーバ 2-2
復号化
Cookie 6-9
説明 9-5
要約 9-3
複数の TMM インスタンス 2-8
複数の要求

開始 6-9
プライオリティ値、割り当て 4-14
プライオリティメンバのアクティベーション 4-14
プライベートクラス IP アドレス
パブリックへのマッピング 13-1
ブラウザ
シャットダウンアラート 9-23
信頼できる CA 9-28
データ圧縮 6-11
ブラウザの回避策 6-20
ブラウザの制限
ネットワークマップ 1-14
フラッド
抵抗 8-22
プラットフォームの要件、MSRDP パーシステンス 7-14
フローティング IP アドレス
SNAT 14-5
プロキシサーバ 2-6
プロキシサーバのキャッシュ 7-10
プロトコル
パーシステンスの設定 7-17, 7-19
バーチャルサーバ 2-15
プロトコルバージョン
指定 9-16, 9-19
設定 9-19
プロトコルプロファイル
一覧 5-1, 5-2
プロファイル
親の作成 5-4
カスタムの作成 5-4
説明 5-1, 9-11
定義 1-12, 5-1
デフォルトの使用 5-3
バーチャルサーバ 5-12
バーチャルサーババ 2-15, 5-12
バーチャルサーバとの関連付け 1-12
プロファイル設定、オーバーライド 9-12, 17-14
プロファイルタイプ 5-1
プロファイルの依存関係 5-13
プロファイルの要約 5-5
プロファイル名、書き換え 6-8, 6-9
プロファイル名、指定 6-4, 6-41

へ

ベーススループットレート 16-5
ベースパケットレート 16-6, 16-8
ヘッダ
クライアント承認 9-6
検査 8-10
挿入 6-6
ヘッダコンテンツ、消去 6-6
ヘッダ挿入
Cookie パーシステンス 7-8
クライアント側の認証 9-27
ヘッダ挿入構文 6-5, 6-6
ヘッダの検索 17-1
ヘルスモニタ
設定 B-10
透過モード 4-8, 4-18
プール 4-7
論理グループ 4-8, 4-18
ヘルプ、オンライン 1-25
変換アドレス
SNAT のための選択 14-9
パーシステンス 7-20
変換アドレスプロパティ 14-11

ほ

ポート固有のワイルドカードバーチャルサーバ、作成 2-12, 2-13
ポート番号
Cookie 7-9
書き換え 6-8
リダイレクト 6-4
ポート変換
NAT 13-2
ポート変換、オフにする 2-12, 2-13
ポート変換設定 2-17
保管域
バースト保管域を参照
ホスト IP アドレスデータグループ 17-17
ホストバーチャルサーバ
定義 2-5
ホスト名、リダイレクト 6-4
補足情報
Platform Guide 1-22
TMOS Management Guide for BIG-IP Systems 1-22
設定用ワークシート 1-22

ま

マスカレード攻撃
抵抗 8-22
マスク、シンプルパーシステンス 7-19
マップ
ネットワークの確認 1-16

み

未使用の帯域幅
借用 16-8
補給 16-6
保存 16-6

む

無効なプロトコルバージョン、設定 9-19

め

メソッド、キャッシュ 6-29
メタデータ、外部データグループ 17-19
メモリメトリック、収集 A-53
メモリレベル gzip アッシュク 6-19

も

文字列照合 6-11, 8-10
文字列データグループ 17-18
モニタ
管理 12-17
ノード 3-5

プール 4-7
モニタ関連付け、タイプ 12-15
モニタとプールの関連付け、管理 4-27
モニタのインスタンス、有効化、無効化 12-17
モニタの設定 12-4

ゆ
ユーザ定義のメトリック、収集 A-53
ユーザ名を抽出、検索方法 10-22
ユーザロール
SNAT プール 14-7, 14-10
レートクラス 16-2
ユニバーサルパーシステンス、定義 7-21
ユニバーサルプロファイルの設定 7-21

よ
要求、チャンクおよびチャンク解除 6-6
予約済みキーワード 12-9, A-1

ら
ラストホッププール、バーチャルサーバ 2-18

り
リソース、制御 9-6
リターンパス 14-2
リダイレクション
書き換え 6-8
定義 6-4
プールの選択 17-5
リダイレクション書き換え
有効化 6-8
例 6-8
リモート認証
TMM サービス 10-1

る
ルータ 2-6
ルーティングの問題 14-3
ルートアドバタイズ 2-20
ルート挿入 2-20
ルール
iRule を参照
ルール演算子、リスト 17-3

れ
レートクラス
管理 16-11
作成 16-2
定義 16-1, 16-2
名前付け 16-5
方向 16-8
割り当て 16-2
レートクラス設定 16-4
レートクラスの例 16-7
レートシェーピング、定義 1-5, 16-1
レスポンダ
CA 10-31
選択 10-30
レスポンダ URL パラメータ 10-30
レスポンダオブジェクト、定義 10-2, 10-23
レスポンダ認証局パラメータ 10-31
レスポンダ定義、選択 10-30

ろ
ローカルトラフィック管理、定義 1-1
ローカルトラフィックサマリ 1-14, 1-15
ロードバランシング、概要 1-7
ロードバランシングバーチャルサーバ 2-1
ロードバランシングプール 1-12, 4-1
ロードバランシング方式
CMP 2-9
定義 4-1
デフォルト 4-11
ロールベースの LDAP 承認 10-23
ロールベースの承認 10-23
論理演算子、リスト 17-3

わ
ワイルドカードサーバ
VLAN への割り当て 2-7
作成 2-12
ワイルドカードバーチャルサーバ、定義 2-6